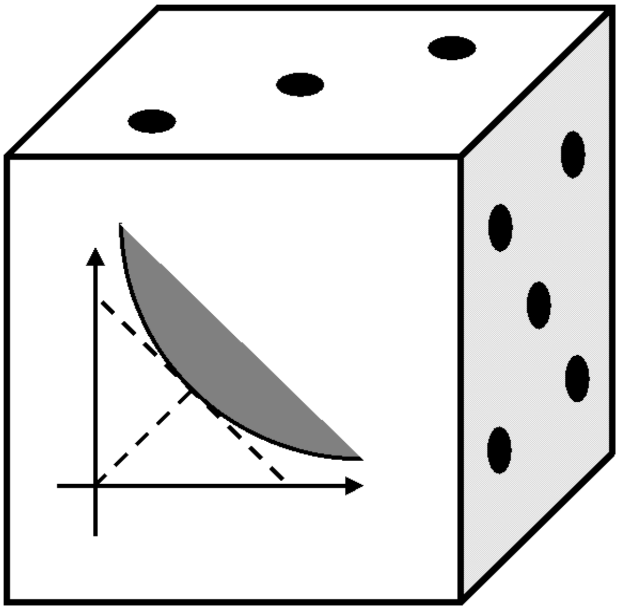
SESAM

USER MANUAL

Proban



General Purpose Probabilistic Analysis Program

DET NORSKE VERITAS

SESAM
User Manual

Proban

General Purpose Probabilistic Analysis Program

Octber 1st, 2004

Valid from program version 4.4

Developed and marketed by
DET NORSKE VERITAS

DNV Software Report No.: 92-7049 / Revision 5, November 1st, 2004

Published by:

Det Norske Veritas
Veritasveien 1
N-1322 Høvik
Norway

Telephone:	+47 67 57 99 00
Facsimile:	+47 67 57 72 72
E-mail, sales:	software.sesam@dnv.com
E-mail, support:	software.support@dnv.com
Website:	www.dnv.com

Table of Contents

1 INTRODUCTION

1.1 Proban - Probabilistic Analysis Program

Proban is a tool for general purpose probabilistic analysis.

The main objective of Proban is to provide a variety of methods aimed at different types of probabilistic analysis. This includes probability analysis of events, distribution analysis, first passage probability analysis and crossing rate analysis.

Proban can deal with a broad class of probabilistic and statistical problems encountered in, for example, engineering and economies.

Proban allows efficient modelling of random variables and events. On-line definition of functions is available.

Proban may be run in batch mode, from a tty-terminal or from a graphics work-station using a modern graphics interface. The same command interface is supported in all modes, and commands generated in the graphics mode are logged and can be read into the program in the line input mode during a later run.

Proban supports a database that contains the input model and results, as well as a journal file that stores a record of all actions done during a program session.

Proban is ideally suited to structural reliability analysis. It may often be convenient to use loads, body motions, or stresses computed by other modules in the SESAM system as input to the reliability analysis.

This manual is valid from Proban Version 4.4.

1.2 Proban in the SESAM System

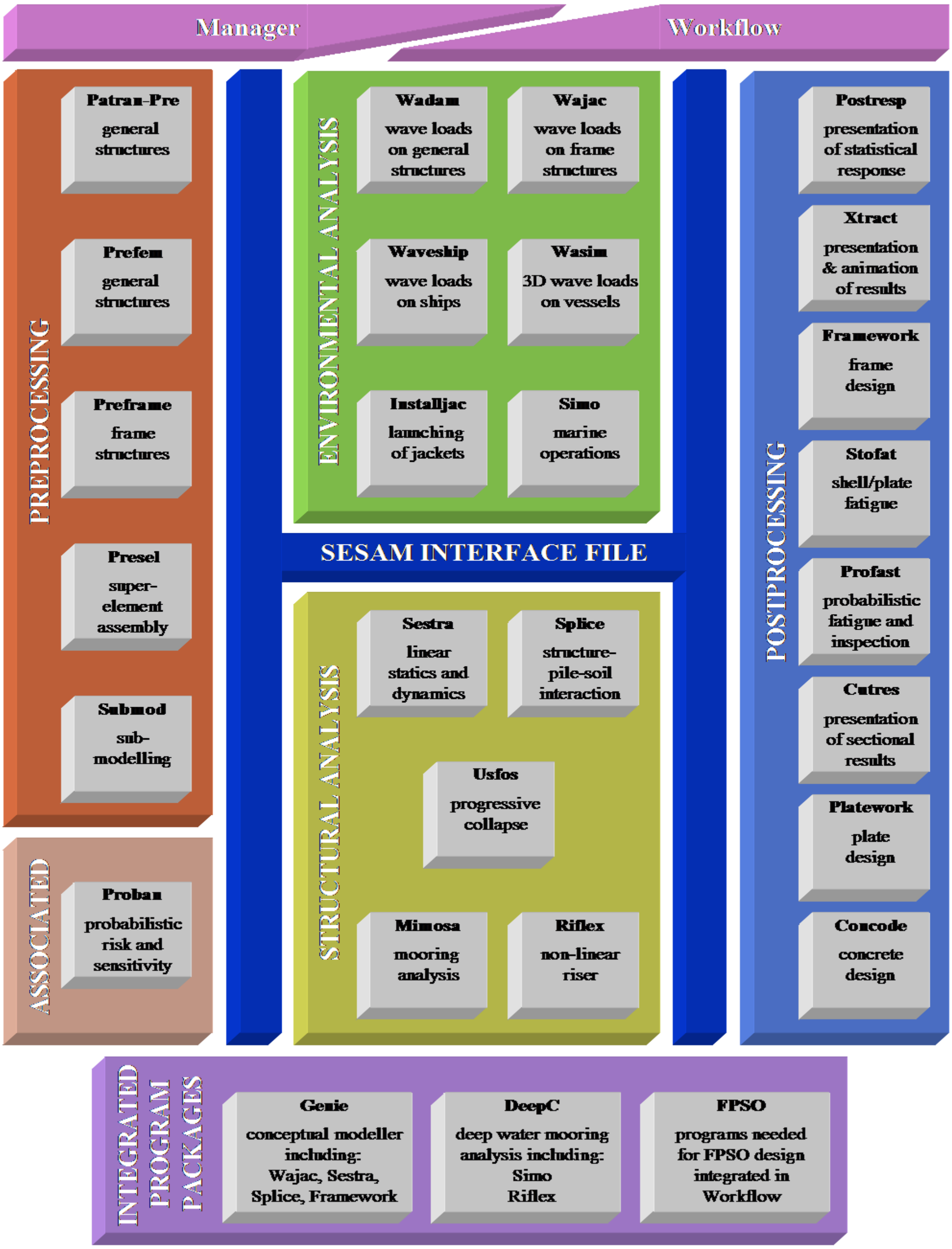


Figure 1.1 SESAM overview

SESAM is comprised of preprocessors, environmental analysis programs, structural analysis programs and postprocessors. An overview of SESAM is shown in Figure 1.1

1.3 How to Read this Manual

Chapter 2 FEATURES OF PROBAN describes the features of Proban, i.e. what the program can do.

Chapter 3 USER’S GUIDE TO PROBAN is the user's guide. It contains guidance on how to exploit the features of Proban.

Chapter 4 EXECUTION OF PROBAN describes how to start the program and how to navigate the user interface. It also describes the files used by Proban and the program’s requirements and limitations.

Chapter 5 COMMAND DESCRIPTION provides a description of all commands and associated input data.

Appendix A PROBAN — LINK IN FUNCTIONS AND DISTRIBUTION explains how to link in functions and distributions defined and coded by the user.

The distribution models available are described in detail in SESAM User’s Manual: Proban Distributions, DNV SESAM Report NO.94-7089/Rev 1, June 1996

The theory is described in detail in SESAM Theory Manual: Proban No. 96-7017/Rev 0,29 September 1996.

1.4 Changes from the Previous Revision

The following changes have been made with respect to the previous revision of the manual (generally described):

- Distribution simulation of vector variables.
- Simple response surface for functions.
- Moment fit of distributions.
- New functions.

2 FEATURES OF PROBAN

2.1 General Description

The overall scope of Proban is to be a practical, software tool for probabilistic analysis.

Proban has a flexible input module, allowing for definition of simple models as well as sophisticated models with complicated dependencies. Proban also has a number of calculation methods available, giving a wide range of results on probabilities, crossing rates, distributions and sensitivities.

This chapter goes through features of Proban in the order in which they would normally be used during a Proban analysis.

The first step in a Proban analysis is to define the question(s) to be answered, and the model that is going to provide the answers. The questions that Proban can answer are typically:

- What is the probability that a given event happens?
- What is the first passage probability of a stochastic process in a given time?
- What is the crossing rate of stochastic process out of a given domain at a specified time?
- What is the mean (standard deviation, skewness or kurtosis) of a given variable?
- What is the distribution of a given variable?
- How much will this result change if an input parameter or value is changed by a given amount?

The second step in a Proban analysis is to define the necessary model functions and code them and link them to Proban. This step may be skipped if the model functions are already available in Proban or can be constructed from the functions that are already available (by use of the function formula facility when necessary).

The third step is to define the model for Proban either interactively or by reading a command input file. The input may be verified through print and display.

The fourth step is to set up and run the analysis. The results may be inspected using print and display/plot.

Proban keeps input and results on a reusable database, so it is possible to exit and restart the program and still have the model and results available.

2.2 Model Definition

A Proban model consists of the following objects:

- Variables
- Events
- Extreme values
- Correlations
- Time derivatives
- Measured values
- Model Functions

Variables, events and functions are referenced by name.

2.2.1 Variables

Variables are the basic building blocks of the Proban model. The term covers traditional random variables as well as variables with a constant value. A variable may be defined as one of the following types:

Fixed	A fixed variable contains a numeric value that is substituted for the variable whenever it is used.
Distribution	A distribution variable is assigned one of the distributions that are available in Proban. Each parameter in the distribution may be defined as a constant value or be assigned an existing random variable.
Fitted-Distribution	A fitted-distribution variable is assigned one of the distributions that are available in Proban by use of distribution fit on observations, on fractiles, on the results of a Proban distribution analysis or on the results of a Proban parameter study on probability.
Function	A function variable is assigned one of the functions that are available in Proban. The function is either created interactively or coded and linked into Proban. Each argument in the function may be defined as a constant value or be assigned an existing random variable.

Generated — A generated distribution variable is assigned the distribution defined by another random variable as its distribution type. The distribution type may be conditioned on values of variables in the definition of the other random variable.

Probability — A probability variable is assigned the probability of an event, possibly in terms of the corresponding reliability index or log probability. The probability may be conditioned on the values of selected variables in the event model.

Time — A time variable is the time parameter of a time dependent stochastic process. It permits time to be an explicit parameter of a probabilistic model.

A great flexibility is obtained, in that a variable can be used as argument or parameter in another variable.

Most distributions in Proban allow for several ways to define the parameters in the distribution (called input sequences), e.g. a normal distribution may be defined through the mean and standard deviation or through the mean and coefficient of variation. The available distributions and input sequences are listed in Section 3.9.1. It is also possible to add user defined distributions.

A multidimensional variable is defined as a multidimensional distribution or as a multidimensional function. These can be referenced directly when a multidimensional value is required, but more often the one-dimensional coordinates are used.

Coordinates in a multidimensional distribution variable are referenced by adding the coordinate number to the variable name after a hyphen (-). Coordinates in a multidimensional function variable are referenced similarly, but by adding the function coordinate name instead of the coordinate number.

The following example illustrates these naming conventions by using a Network function. This function has been programmed as a multidimensional function in Proban. The names of the function coordinates are: Path1, Path2, Path3 and Longest.

```
CREATE VARIABLE X ' ' DISTRIBUTION Multi-Normal 7 <parameters>
CREATE VARIABLE F ' ' FUNCTION Network X-1 X-2 X-3 X-4 X-5 X-6 X-7
RUN DISTRIBUTION-ANALYSIS F-Path1
```

2.2.2 Events

Calculation and examination of probability is often the goal of a Proban analysis. The probability is associated with an event in the input model, for example the event that a Net Present value is negative, or the event that at least one of three components in a series system fail.

There are four different types of events in Proban:

Single — A single event is the event that a value of a variable is less than, equal to or greater than a numerical threshold value. The single event is the basic event in Proban.

Intersection — An intersection event is an intersection of other events, i.e. it is fulfilled only when all subevents are fulfilled. All events except conditioned events may be subevents in an intersection.

Union	A union event is a union of other events, i.e. it is fulfilled if at least one of the subevents is fulfilled. All events except conditioned events may be subevents in a union.
Conditioned	A conditioned event facilitates analysis of conditional probability. It has two subevents: the event that is conditioned and the event condition on. All events except conditioned events may be used to define a conditioned event.

As with variables, this provides for a great flexibility in definition of events. Unions and intersections can be built on top of each other freely, defining a complex network of events if required.

2.2.3 Extreme Values

A uni-variate random variable with one of the type attributes: Distribution, Fitted-distribution and Generated can have its definition replaced by the maximum or minimum of an integer number of independent identical realisations of the variable.

Notice that the distribution parameters are kept fixed when the extreme value is taken. (In case of a generated distribution the variables conditioned on serve as distribution parameters.)

2.2.4 Correlation

Correlations are used to model linear dependency between variables. Two variables will have a positive trend (usually becoming large together and small together) if their correlation is positive. When the correlation reaches the maximal value of one, they become linearly dependent. Conversely, the variables have a negative trend (one is usually large when the other is small) if their correlation is negative, and they again become linearly dependent when the correlation reaches the lower limit of -1. Note that two variables may be dependent on each other in a nonlinear way and at the same time have correlation coefficient equal to zero. In such a case a more refined modelling is required. See Figure 2.1.

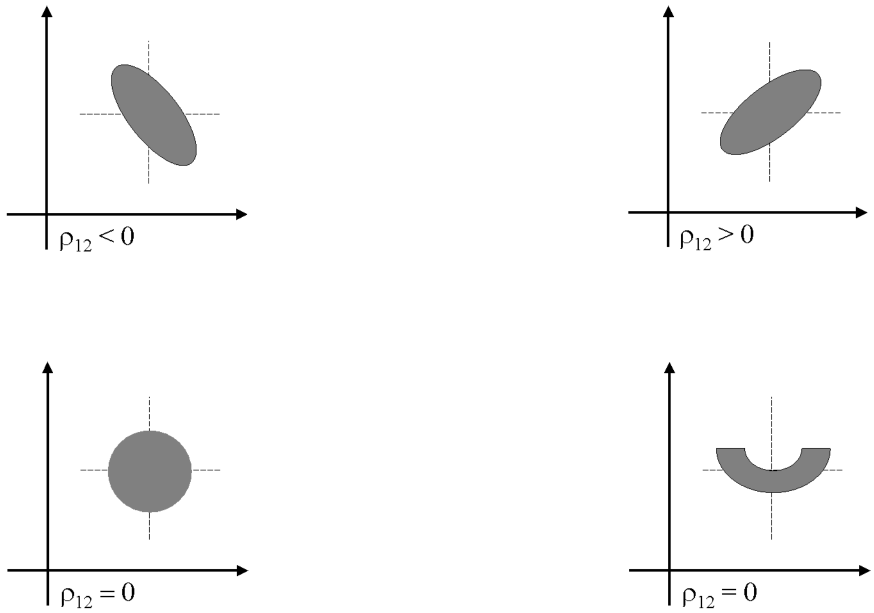


Figure 2.1 Correlations and dependencies between variables

Correlations can be defined between uni-variate variables with type attribute Distribution, Fitted distribution or Generated. In case a variable is a generated distribution the input correlation is the corresponding normal correlation. In all other cases the model space (basic) correlation may be input alternatively.

Proban accomplishes the correlation of non-normal distributions by transforming the variables to standard normal variables (as described in Section 2.3.1) before defining the correlation the usual way between the standard normal variables. This yields the Nataf distribution model which is the natural generalisation of the Multi-Normal distribution to correlation of non-normal random variables.

The Nataf distribution model may define a valid range of a basic correlation coefficient as [-a,b] with a and b strictly less than 1. Illegal basic correlation indicates that non-linear dependency is present in the model and that this is not captured by the Nataf distribution model.

Correlation of normal random variables can also be input by use of a Multi-Normal distribution. This is a multidimensional distribution with normal marginal distributions and a full correlation matrix.

Notice that creating dependencies between variables will introduce correlation. As an example, if both A and B are functions of C, and C is a random variable, A and B will be dependent and most likely also correlated. This provides a means to model statistical dependency that is not captured by the Nataf model.

2.2.5 Time Derivatives

A continuous stochastic process is modelled through variables which represent the stochastic process and their corresponding time derivatives. A process variable and its corresponding time derivative variable are both random variables with the same dimension and with type attributes Distribution, Fitted-distribution or Generated. The mean value of a time derivative variable must be zero.

A continuous stochastic process can be viewed as a particle which moves continuously in time, see Figure 2.11.

2.2.6 Measured Values

A variable may model the measurement of a physical quantity, e.g. the depth of a crack in a beam subjected to fatigue loading. One may wish to calculate the reliability of the beam conditioned on the information obtained by the measurement. A single equality event models the event that the crack has grown to the measured depth and the measured value variable models the uncertainty of the measurement.

It is necessary to attach the measured value variable to the single equality event in order to calculate a correct conditional probability since the calculation depends on this relation. The attachment is specified on input.

2.2.7 Model Functions

Most of the complexity of the model to be analysed is hidden inside the model function. A model function can be coded by the user and linked into Proban or be created interactively as a function formula. In many cases the set of built-in functions together with the function formula facility will be sufficient to build the required model.

Because of the flexibility Proban offers for definition of variables, a basic set of functions provides building blocks from which a great many models can be built. Proban is delivered with the basic arithmetic functions, the basic mathematical and trigonometric functions and a few useful additions to these.

During an analysis Proban usually needs to take derivatives of the model functions. These derivatives may be programmed into the functions in order to enhance performance, or may be left out, in which case Proban will do the differentiation numerically.

A programmed model function returns either a single value or a vector value. A function created as function formula returns a single value.

How model functions are programmed and linked into Proban is described in Section 3.10.3.

How model functions are created interactively is described in Section 3.10.2.

The input model is verified using the PRINT, DISPLAY, PLOT and RUN DETERMINISTIC-ANALYSIS commands.

Newly programmed model functions can (and should) be checked using the PRINT FUNCTION command, which allows for checks of function values and gradients.

Some checks cannot be done before the analysis is initialised or run. Most of these can be done using the RUN INPUT-CHECK command. This command will check the consistency of the model, but not do the actual run. It traps most, but not all, errors.

2.2.8 Generated Distribution

Sometimes the maximum or minimum of a number of independent identically distributed realisations of a function $G(\boldsymbol{X};\alpha)$ of random variables $\boldsymbol{X}$ is required. This is facilitated by creating Z as the Generated distribution of G, conditioning Z on α and assigning the appropriate extreme value to Z.

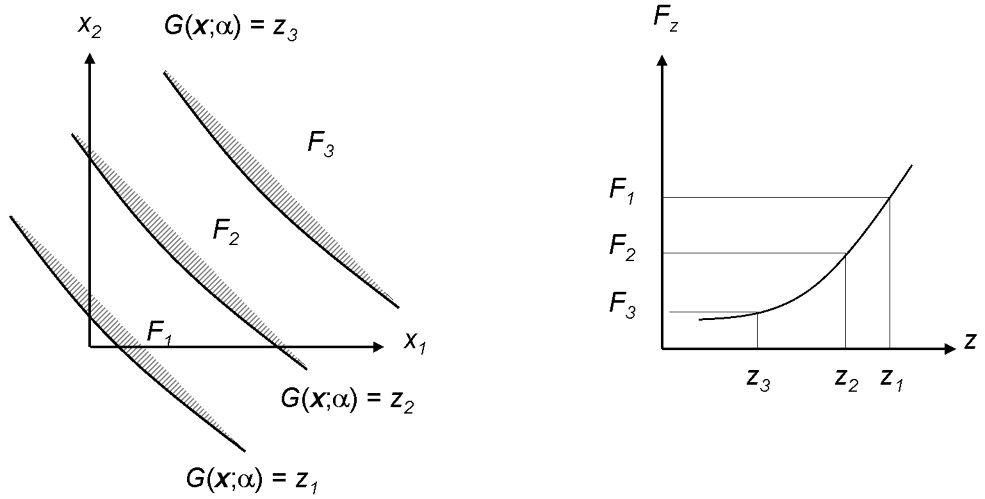


Figure 2.2 Generated Distribution - Distribution of level surfaces

Geometrically the Generated distribution is the distribution F_Z of level surfaces, $z = G(\boldsymbol{X};\alpha)$, of the corresponding function of random variables. The vector α is the current realisation of variables conditioned on. Three points on the distribution are shown in Figure 2.2. The Probabilities and fractiles of a generated distribution are approximated by pointwise application of the FORM method. Because the random variables $\boldsymbol{X}$ are integrated out in the calculation process, the random variable Z is uncorrelated with other variables, unless such correlation is explicitly defined. Dependency on other variables is modelled through the variables conditioned on.

An arbitrary number of generated distributions can be defined.

Random variables having generated distributions can be correlated with other distribution variables.

It is not possible to include a generated distribution or a probability variable in the vector $\boldsymbol{X}$ above since those variables introduce an extra level of optimization.

Calculations of fractile from probability and probability from fractile generally require different optimization algorithms. Optimization criteria and differentiation increments are defined separately for the generated distribution.

2.3 Probability Analysis

Proban supplies several methods for finding the probability of an event and the associated sensitivity results. The methods fall into two categories:

- Analytical methods
- Simulation methods

The analytical methods include FORM and SORM (First and Second Order Reliability Methods). These give approximate results relatively fast, but require that the model functions are differentiable (twice differentiable for SORM). The accuracy of FORM is usually good for small probabilities. The accuracy of SORM is often good over the whole probability range.

Simulation methods take longer time to run than FORM/SORM, but do not put similar demands on the model functions and the distribution functions. Thus the simulation methods provide analysis tools for models other than structural reliability models. Within structural reliability simulation methods are used both to verify and to improve a result obtained from a FORM/SORM approximation and also to obtain results when FORM/SORM cannot be used. The features of the different analysis methods are described below.

2.3.1 FORM/SORM

Calculation of the probability of an event may be formulated as a multidimensional integral (see also the left part of Figure 2.3).

$$P(Event) = \int_{Event} f_X(\boldsymbol{x})d\boldsymbol{x}$$

The variables ***X*** are the distribution variables in the model, fX(x) is their joint probability density function, and the probability is integrated over the domain of x in which the event occurs.

The FORM and SORM methods have been developed with the purpose of approximating this integral. This is accomplished by approximating the surface at the boundary of the area, where the event is fulfilled, in such a way that the integration can be done over the approximated area.

The trick to do this is twofold.

- First the random variables ***X*** are transformed into independent standard normal variables ***U***.
- Secondly, the area where the event is fulfilled is approximated by an area bounded by hyperplanes (FORM) or a second order surface (SORM).

Theoretical results for integration of the standard normal density over such areas can then be applied.

In order to understand the FORM/SORM method, it is necessary to describe the transformation into the standard normal spaces (called *V*-space and *U*-space) first. The description here is not theoretically complete.

During an analysis Proban always operates in a transformed space, where all variables are independent and have standard normal distributions. It is possible (in theory) to map any distribution into such a space using a one-to-one transformation. Proban first maps those input variables, that are defined as distributions, to

standard normal variables in the so-called *V*-space. These standard variables may still be correlated if correlations have been assigned (See Section 2.2.4). The coordinates in *V*-space correspond to variables in the input space. Secondly, the *V*-space is mapped into *U*-space, so that the *U*-space variables are uncorrelated.

The event that is being analysed is formulated inside Proban as: $G(\boldsymbol{x}) < 0$ where G is an appropriate function. The terminology used here derives from structural reliability analysis (calculation of small probabilities):

The function G is called the *limit state function*.

The set where the event is fulfilled is formulated as: $G(\boldsymbol{x}) < 0$, and is called the *failure set*.

The surface where $G(\boldsymbol{x}) = 0$ is called the *failure surface* or the *limit state surface*.

The set where $G(\boldsymbol{x}) > 0$ is called the *safe set*.

These terms and the transformation is illustrated in Figure 2.3.

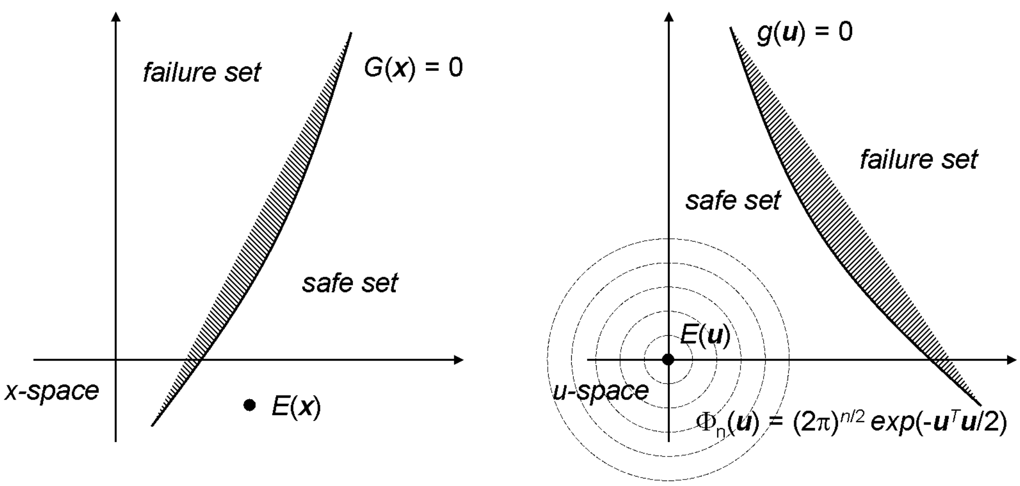


Figure 2.3 Transformation from input space to *U*-space

The required probability: $P(g(\boldsymbol{u}) < 0)$ is approximated using the following steps:

- An approximation point is found using an optimization method - the nearest point to the origin on the failure surface. This point is called the *design point* u*.
- The failure surface is approximated at this point using either a linear approximation (FORM) or a second order approximation (SORM).
- The probability content in the failure set is approximated by the probability content in the approximated failure set. The *Reliability Index*, β, is defined as the standard normal fractile corresponding to the prob-

ability of the safe set. In the simplest case β defaults to the distance from the origin to the design point in U-space.

The process is illustrated in Figure 2.4.

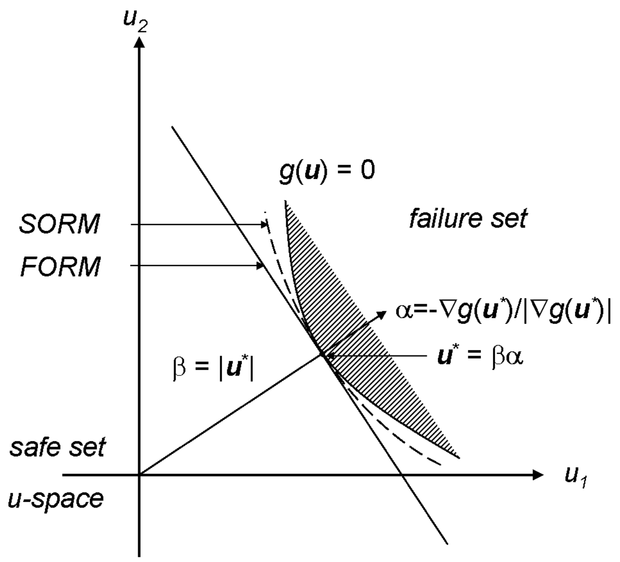


Figure 2.4 FORM/SORM approximation to failure surface

The stop criteria of the optimization method may be controlled. A starting point other than the origin for the optimization may be defined together with bounds on the optimization variables. Analytical differentiation of the model function is used when possible, if this facility has not been turned off. Step lengths for numerical differentiation can be defined.

Using FORM, the failure probability is estimated as the probability outside the linear (hyperplane) approximation to the failure surface. This probability is

$$P_{\mathrm{FORM}} = \Phi(-\beta)$$

where Φ is the standard normal distribution function, and β is the distance from the origin to the design point.

Using SORM, the failure surface is approximated with a second order surface, and the probability outside this surface is calculated. The reliability index in this case becomes a function of the failure probability:

$$\beta = -\Phi^{-1}(P_{\mathrm{SORM}})$$

Various types of second order approximations are available, giving different accuracies and requiring different numbers of second order derivatives.

FORM/SORM can be used on a single event, a union of single events, an intersection of single events or a union of intersections with each intersection containing single events.

Proban handles union and intersection events a little differently than single events, but the basic principle is the same. Unions and intersections may generate two different geometries in *U*-space, the so-called *Large intersection* and *Small intersection*. A large intersection is generated from a small probability in a union event or a large probability in an intersection event. A small intersection is generated in the converse circumstance. The situations are described in Figure 2.5 and Figure 2.6.

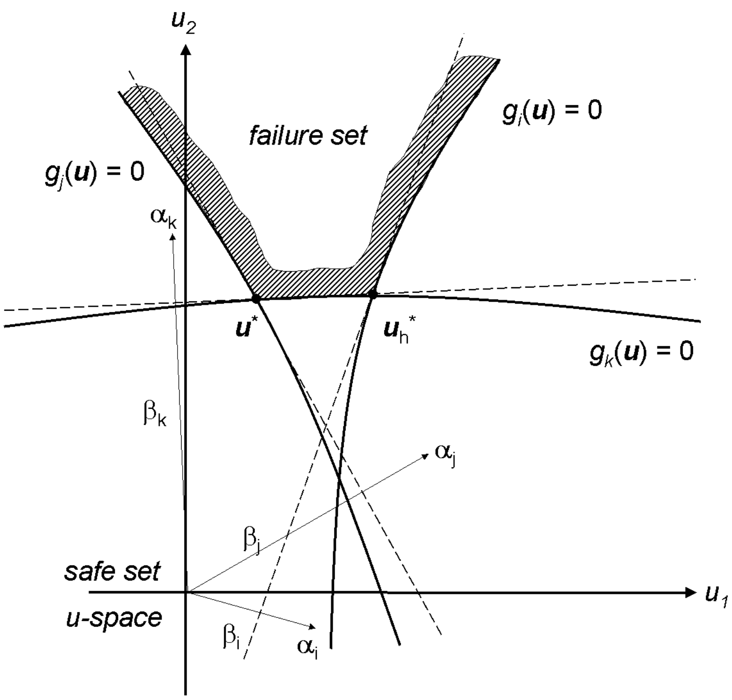


Figure 2.5 Small intersection geometry in a FORM/SORM analysis

In the case of a small intersection, the approximation of the failure set becomes convex and the probability of this set can be calculated directly using known methods for calculating probabilities in the multinormal distribution.

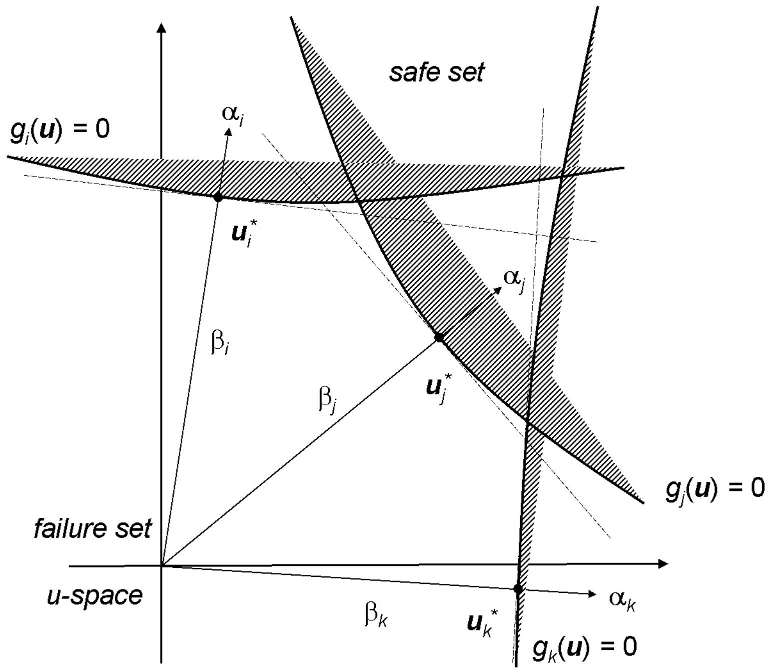


Figure 2.6 Large intersection geometry in a FORM/SORM analysis

In the case of a large intersection, the safe set is the convex set, and the failure probability is calculated as 1 - the probability of the safe set when the direct multinormal probability calculation is used. Alternatively, for cases where the accuracy of the multinormal probability calculation is in doubt (i.e. the probability is close to 0.5), the failure probability can be estimated using bounds.

In some cases, a subevent may be inactive in the first linearisation, because it is partly hidden behind the others. An example is seen in Figure 2.5. Proban will subsequently attempt a separate linearisation of this event in order to obtain the best estimate possible of the probability. However, this linearisation of *inactive constraints* can be turned off if desired.

If a union of intersections is being analysed, Proban will analyse each intersection first, then estimate the total failure probability using the same bounding technique as for large intersections.

2.3.2 Monte Carlo Simulation

Monte Carlo simulation is the simplest simulation method available in Proban. It consists of sampling random points and checking if each point is inside or outside of the event of interest. The probability of the event is estimated as the average number of hits in the event during the simulation.

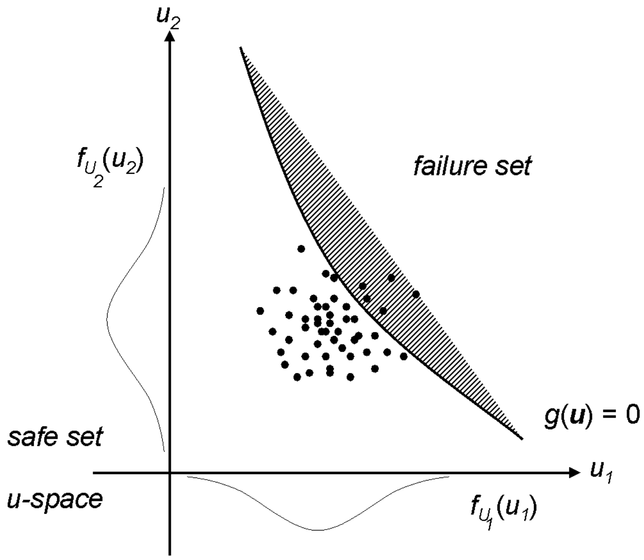


Figure 2.7 Monte Carlo hit/miss simulation of a probability

This method is not efficient, except perhaps for mid range probabilities, or for sufficiently simple model functions, but it has the definite advantage that it will produce unbiased estimates. Thus it may be used to check if the approximate result delivered by other methods is accurate. Its other main use is in cases, where the more sophisticated methods cannot be used, e.g. because the model function is not differentiable.

The length of a Monte Carlo simulation may be controlled by defining the maximal number of simulations, by restricting the time to be used or by demanding a stop when a certain coefficient of variation has been reached.

2.3.3 Directional Simulation

Directional simulation is a sophistication of the principle used in Monte Carlo simulation. The rotational symmetry of *U*-space is used to make sampling more efficient. Instead of sampling points randomly in the *U*-space, directions are sampled randomly, and the probability of the event along the sampled direction is calculated. Because of the rotational symmetry, each directional probability estimate is an unbiased estimate of the correct probability.

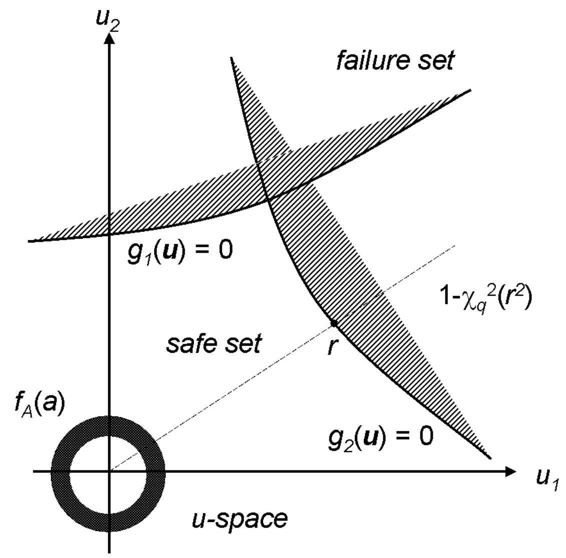


Figure 2.8 Directional simulation

This method is in theory unbiased, but may still produce biased results because it requires correct evaluation of the probability in a given direction, which in turn requires solving a nonlinear equation to find the point(s) where the limit state surface crosses the line. In complicated cases there may be more than one of these points in a given direction, and to be sure to find them all will cost computation time. Proban comes with three different search methods, giving different trade-off between speed and accuracy.

There are ways to further sophisticate this sampling. Proban always samples also the probability in the opposite direction of any given direction. A set of orthogonal directions, spanning the whole space, may be sampled instead of just one direction, and linear combinations of these may be considered. This still provides unbiased probability estimates because it utilises the rotational symmetry of the *U*-space. However, the required time to produce a single estimate of the probability increases considerably with the number of random variables in the problem, so the sophisticated methods are not recommended for problems with many variables. Proban supplies a default method, that is efficient in most cases.

The length of a Directional simulation my be controlled by defining the maximal number of simulations, by restricting the time to be used or by demanding a stop when a certain coefficient of variation has been reached.

2.3.4 Axis Orthogonal Simulation

Axis Orthogonal simulation is also a directional simulation technique. However, instead of shooting from the origin as in Directional simulation, it shoots from a hyperplane based on a FORM approximation out towards the limit state surface. Axis Orthogonal simulation does not simulate the probability itself, it simulates a correction to the FORM approximation to the probability.

An example of a small intersection is being analysed is shown in Figure 2.9. The method cannot be applied to large intersection geometries.

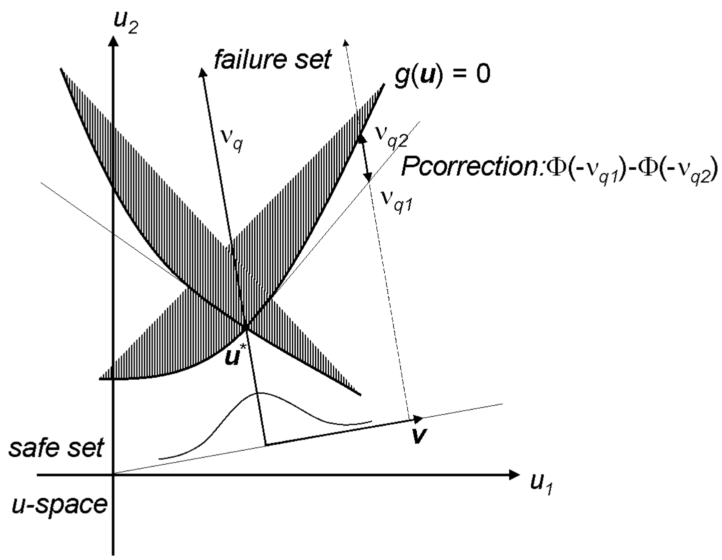


Figure 2.9 Axis Orthogonal simulation

The simulation consists of sampling points on the hyperplane, that is "perpendicular" to the limit state surface, and then finding the correction to the failure probability along a line perpendicular to the hyperplane and originating from the sampled point.

There are two ways to sample points on the hyperplane. A standard normal density may be used or a conditioned sampling density, taking the shape of the limit state surface into account, may be used. The standard normal sampling density will give a simulated *additive* correction to the FORM probability, while the conditioned density will give a *multiplicative* correction to the FORM probability.

As in Directional simulation, a nonlinear equation must be solved in order to find the point(s) where the failure surface intersects the sampled search direction. Three search methods are supplied, giving different trade-off between safety and speed.

The length of an Axis Orthogonal simulation my be controlled by defining the maximal number of simulations, by restricting the time to be used or by demanding a stop when a certain coefficient of variation has been reached.

2.4 Nested FORM Analysis

The nested FORM analysis is invoked when a model contains a probability variable or when a model contains both a continuous stochastic process and at least one other distribution variable.

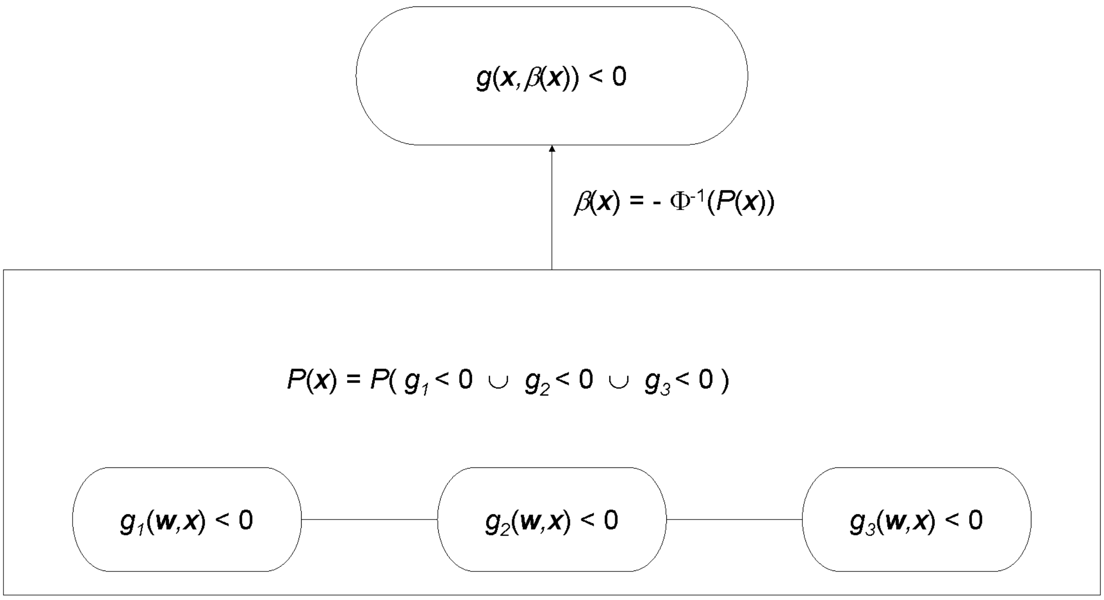


Figure 2.10 Nested reliability analysis

The outer integration level is a design point search for the single event which contains the probability variable. Variables conditioned on are integrated on the outer level together with variables not contained in the event of the probability variable.

The inner integration level calculates the (log)probability (or reliability index) of the event of the probability variable. The result is calculated given the current values of variables conditioned on.

Figure 2.10 shows a nested FORM analysis resulting from a single event model which involves a probability variable, the probability variable itself being the probability of a union event of single events.

The outer loop event is always a single event model.

Proban checks for inconsistencies in the separation between outer integration level and inner integration level resulting from inconsistent selection of conditioning variables.

The calculation method available is FORM on both levels. Optimization options and differentiation increments are defined for the outer integration level and the inner integration level separately.

2.5 First Passage Probability Analysis

A *first passage probability* is the probability that a continuous stochastic process initially is in the failure set defined by an event plus the probability that it starts in the safe set and enters the failure set at least once within a specified time interval. This is shown in Figure 2.11 for the process $\boldsymbol{X}(t)$ with starting time T_S=0 and duration $D > t$.

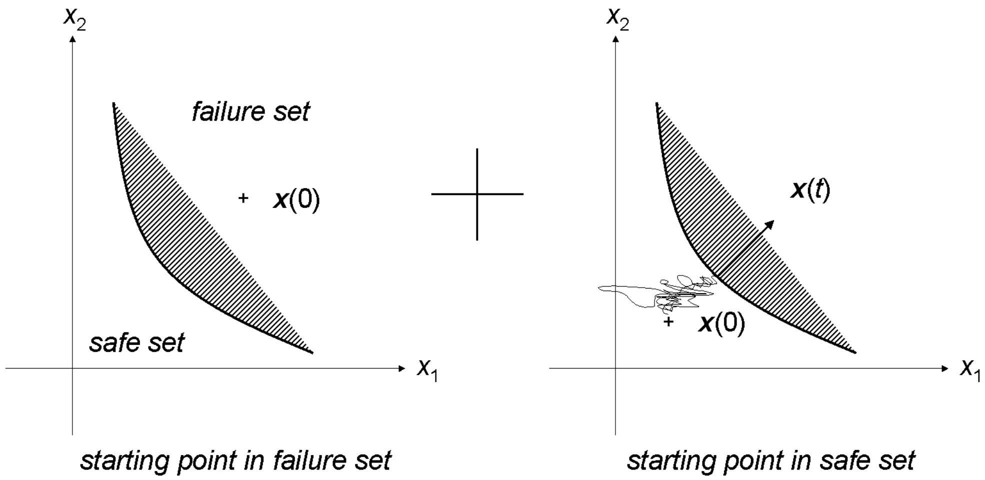


Figure 2.11 First Passage Probability

The continuous stochastic process is modelled by assignment of a time derivative process variable to the process variable.

The time interval is modelled with starting time T_0 and duration D. When a variable with type attribute time is a part of the model, T_0 and D must be attached to this variable or be defined as defaults. If no time variable is present in the model, only the default duration is required. The starting time and the duration may be modelled as random variables.

Notice that if an ordinary probability analysis (not first passage probability analysis) is carried out on a model which includes a stochastic process, then the time derivative variables are neglected and the time variable is replaced by its corresponding starting time.

The calculation of a first passage probability has two steps. Firstly the probability that the event is fulfilled at the starting time is calculated. Secondly, the expected number of crossings is calculated. Then these two results are combined by the Poisson formula to give the first passage probability.

$$P_{\text{First Passage}} = 1 - \Phi(\beta)\exp\left(-\int \nu(t)dt\right)$$

in which β is the reliability index corresponding to the probability that the process is in the failure set at time T_0, $\nu(t)$ is the mean crossing rate at time t and the integral is taken over the interval $[T_0, T_0 + D]$.

If a time variable is present in the model, then the time interval is integrated over in order to calculate the time averaged mean crossing rate. Time integration is carried out by use of a trapezoidal quadrature. The number of quadrature points is specified on input. A reduced integration interval may be specified in order to calculate only significant contributions. Periodicity in the process may be exploited to further reduce the integration effort.

Proban divides the random variables into two sets. Those variables which describe the time dependent stochastic process constitutes Set A and the remaining random variables constitutes Set B. Set A is integrated over to give the first passage probability for the stochastic process conditioned on the values of the variables of Set B. The outer integration level averages this first passage probability over the variables of Set B. The implied nested optimization employs the optimization criteria defined for nested reliability analysis (see above).

If a random variable which is not a time dependent stochastic process is to be integrated at the inner integration level, then this is achieved by pushing the variable to the inner level.

2.6 Crossing Rate Analysis

The rate $\nu(t)$ of a continuous stochastic process crossing into a failure set at time t is calculated as a parallel system sensitivity measure employing the FORM method.

The continuous stochastic process is modelled as explained for first passage probability calculation.

If a time variable is present in the model, then the time interval is averaged over in order to calculate the time averaged mean crossing rate. Time integration is carried out by use of a trapezoidal quadrature. The number of quadrature points is specified on input. A reduced integration interval may be specified in order to calculate only significant contributions. Periodicity in the process may be exploited to further reduce the integration effort.

Proban divides the random variables into two sets. Those variables which describe the time dependent stochastic process constitutes Set A and the remaining random variables constitutes Set B. Set A is integrated over to give the (time averaged) mean crossing rate for the stochastic process conditioned on the values of the variables of Set B. The outer integration level averages this crossing rate over the variables of Set B. The implied nested optimization employs the optimization criteria defined for nested reliability analysis (see above).

If a random variable which is not a stochastic process is to be integrated on the inner integration level, then this is achieved by pushing the variable to the inner level.

2.7 Distribution Analysis

In many cases, the distribution of a random variable is of interest. Proban supplies three

different ways of calculating this distribution.

The Mean Value Based FORM method is analytical, though not very accurate. The two simulation methods, Monte Carlo simulation and Latin hypercube simulation, are recommended for use when possible.

2.7.1 Monte Carlo Simulation

Monte Carlo simulation is a straightforward simulation technique, where points are sampled randomly and the target value is calculated each time. The sample of target values is stored on the database and can be used for display or printed presentation after the analysis. The first four moments of the distribution are fitted from the sample and presented as a summary result after the analysis is complete.

The first four moments are illustrated in Figure 2.12 to Figure 2.15. A normal distribution has a skewness of 0.0 and a kurtosis of 3.0. A lognormal distribution has a positive skewness and a kurtosis that is larger than 3.0.

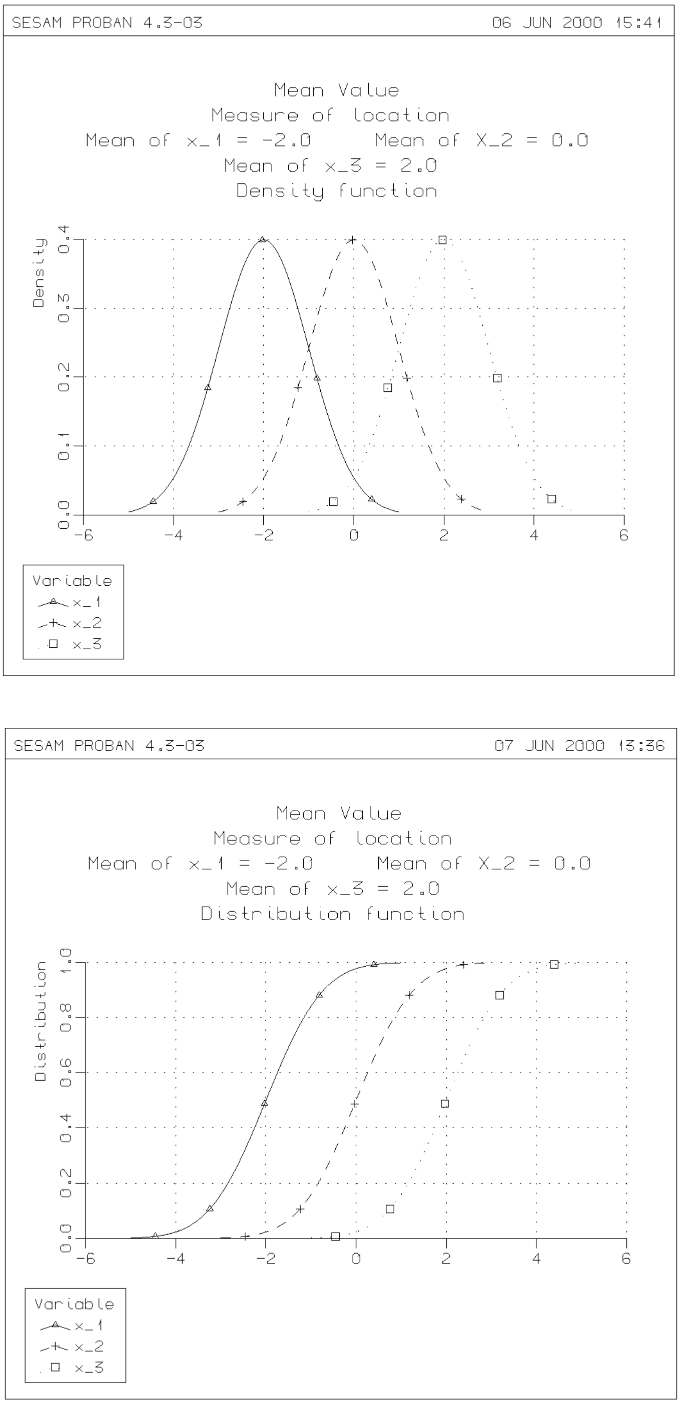
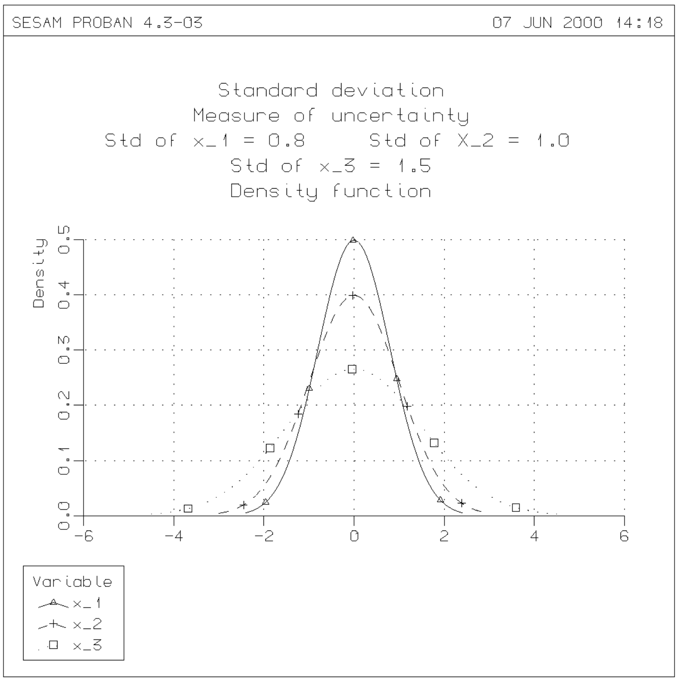


Figure 2.12 Illustration of Mean



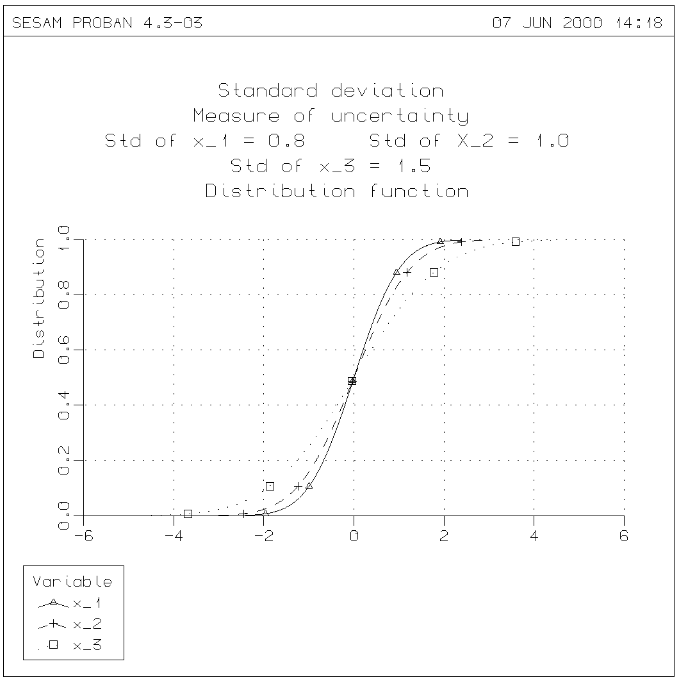


Figure 2.13 Illustration of Standard Deviation

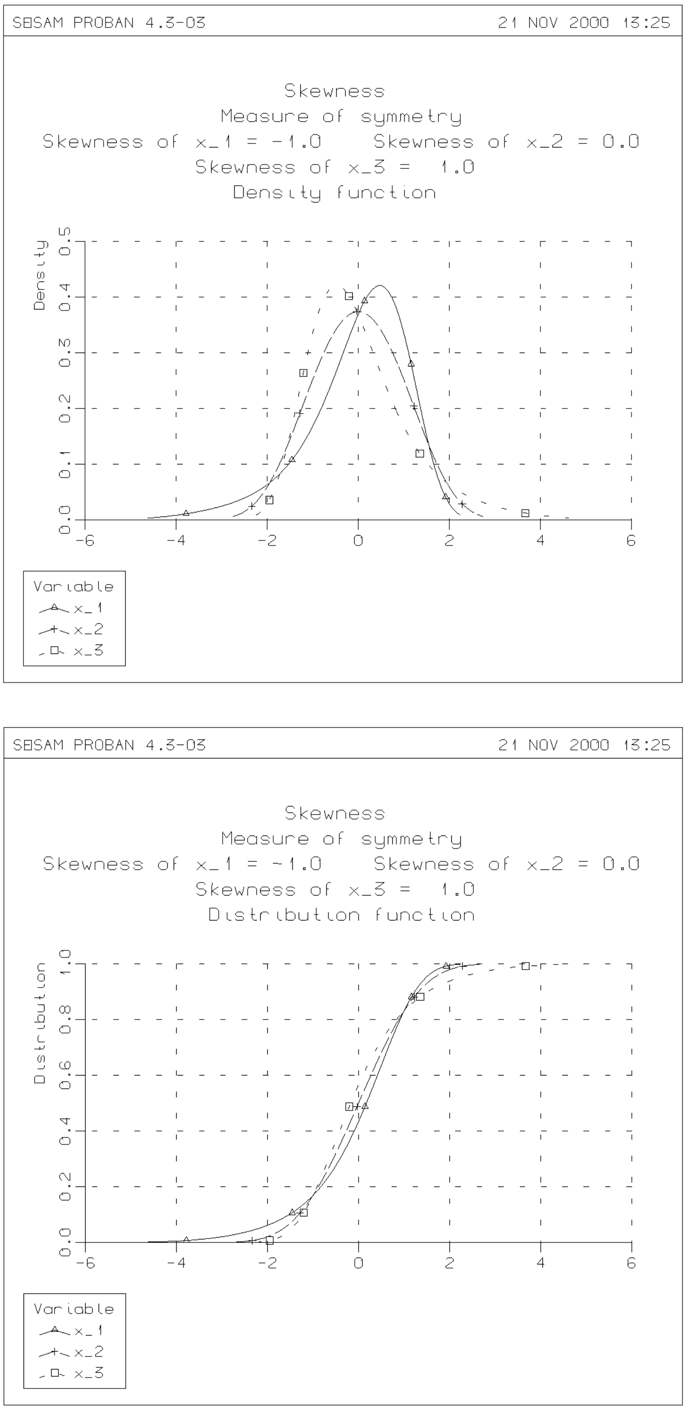


Figure 2.14 Illustration of Skewness

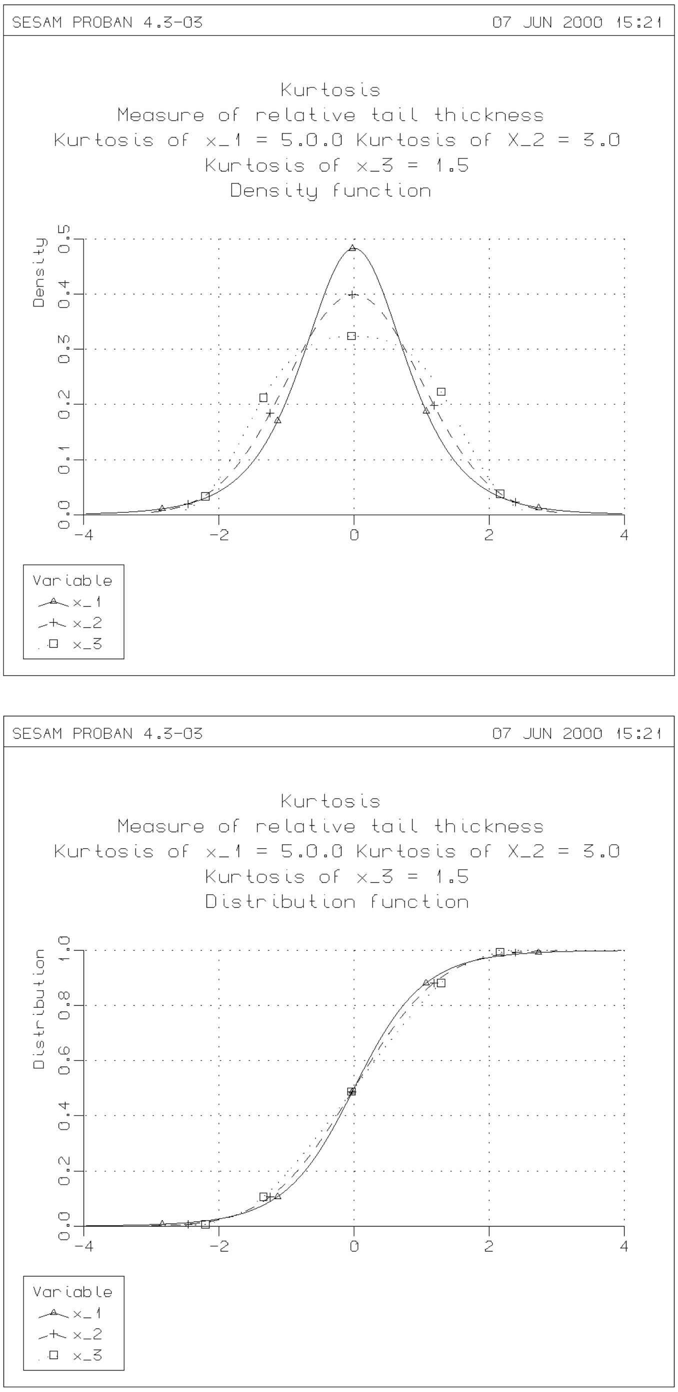


Figure 2.15 Illustration of Kurtosis

Proban fits (if possible) a Hermite transformation distribution to the sample, using the estimates of the first four moments. This is stored in a variable called Hermite-Fit. Proban will also fit a normal distribution, using the estimated mean and standard deviation. This is stored in a variable called Normal-Fit.

It is also possible to fit other distributions to the sample by creating variables with type attribute Fitted Distribution, see Section 3.9.2.

2.7.2 Latin Hypercube Simulation

Latin hypercube simulation is a refinement of Monte Carlo simulation, designed to be used in cases where the calculation of a sample point is time consuming. The sample points are spread out over the sample space in a systematic way in order to cover the space as well as possible with a few points. The technique is illustrated in Figure 2.16.

Each axis is divided into a number of intervals (the number of intervals being equal to the number of sample points). Each of these intervals has the same probability content. One coordinate is sampled from each interval on each axis, and the coordinates are combined into sample points in the sample space in such a way that each coordinate is used exactly once. This ensures a spread of the points over the sample space.

The target value is calculated in each of these points, and these values are then treated as an ordinary sample, as described in Section 2.7.1 above.

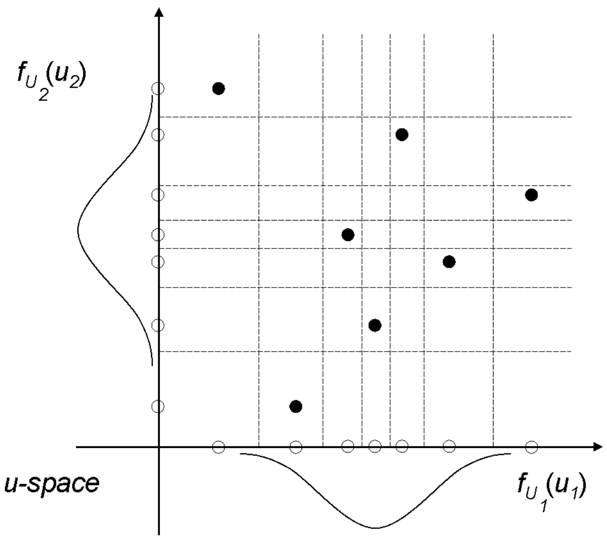


Figure 2.16 Latin hypercube simulation

Proban fits (if possible) a Hermite transformation distribution to the sample, using the estimates of the first four moments. This is stored in a variable called Hermite-Fit. Proban will also fit a normal distribution, using the estimated mean and standard deviation. This is stored in a variable called Normal-Fit.

It is also possible to fit other distributions to the sample by creating variables with type attribute Fitted Distribution, see Section 3.9.2.

2.7.3 Mean Value Based FORM

Mean value based FORM is often an unreliable method, and the only reason for including it in Proban is that it is fast. It allows estimation of distributions that cannot be simulated because of extreme computation times. Another useful application of the method is to quickly identify the range of a distribution.

The principle of Mean value based FORM is to estimate the distribution from the FORM approximation of the limit state level surface through the origin of *U*-space. If $g(\mathbf{0})$ is the value at the origin, the probability that $g(\mathbf{u}) < g(\mathbf{0})$ is estimated as 0.5 using FORM (the reliability index β is 0). Shooting out in the direction of the *U*-space gradient at the origin, the function value at points along the gradient direction are related to the distance β (reliability index) from the origin, and β is in turn related to the corresponding probability, Φ $(-\beta)$. The error made can be small and can be very large. It depends on the angular difference between the gradient at the origin and the direction to the correct design point $\boldsymbol{u}^*$ in Figure 2.17.

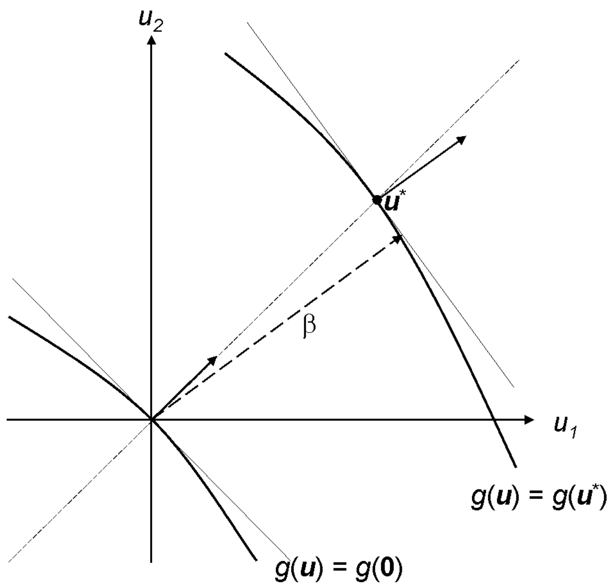


Figure 2.17 Mean value based FORM

In extreme cases, Mean value based FORM may give an estimated distribution function that is not increasing everywhere, so the method must be used with caution.

It is possible to fit other distributions to the sample by creating a variable with type attribute Fitted Distribution, see Section 3.9.2.

2.8 Sensitivity Results

It is often desirable to investigate the sensitivity of a target value with respect to one or more parameters in the model. The target value can be the calculated probability or reliability index, or the moments of a simulated distribution.

Examples are the sensitivity of the reliability index with respect to the standard deviation of the strength of a material, or the sensitivity of the mean and standard deviation of the Net Present Value of an investment with respect to the oil price.

The change in e.g. the reliability index given a change in a parameter τ is estimated as:

$$\beta_{new} = \beta + \frac{d\beta}{d\tau}\Delta\tau$$

Proban can calculate the sensitivity of the target value with respect to any fixed variable or constant distribution parameter or constant function argument in the model.

The sensitivity of the probability and reliability index with respect to a parameter can be calculated using FORM/SORM and Directional simulation.

The sensitivity of the mean, standard deviation, skewness and kurtosis of a distribution can be calculated using Monte Carlo or Latin hypercube simulation of a distribution.

The derivative of a target value with respect to a parameter is not very easily compared to a derivative with respect to another parameter, because the two parameters may have values of different magnitude. For this reason, Proban uses a concept called a sensitivity measure in order to quantify sensitivity values on the same scale.

The sensitivity measure is defined as the change in the target value estimated from a fixed relative increase in the parameter (the default increase is 10%). This value will have the same scale as the target value, independent of the scale of the magnitude of the parameter used. The sensitivity measure is not properly defined if the parameter value is very close to 0, thus a limit at which it is applied must be set. The relative increment and the limit at which it is applied are controlled using the DEFINE RESULT OPTION command.

Another kind of sensitivity that is of interest is the degree of importance the uncertainty of a random variable in the model has on the probability or reliability index. This can be used to identify those random variables in the model that could just as well be fixed (at the 50% fractile), and to identify those random variables for which it would pay to reduce the uncertainty (if possible).

These sensitivities are presented in Proban as *importance factors*. They are presented in % and will always sum to 100%.

The usage of importance factors can be illustrated by the following example. If a variable has the importance factor α (in %), the effect on the FORM reliability index of fixing the variable to a constant value is estimated to be:

$$\beta = \frac{1}{\sqrt{1 - \frac{\alpha}{100}}}$$

Please note that this formula applies only to FORM analysis of single events.

If two or more variables are correlated, only one importance factor will be presented for the group. The same applies to distribution variables where one variable enters the distribution of another variable as a parameter.

Importance factors can be calculated using FORM/SORM and Directional simulation.

2.9 Deterministic Analysis

It is often desirable to evaluate the value of a variable or an event function at a given point. This is achieved through performing a *deterministic* analysis. The analysis of a variable can be done at the mean value or at the median value of the stochastic variables involved, or at a point modified from one of these. The analysis of an event function can be done at the u-space origin or at the starting point for a FORM/SORM analysis. Thus, in order to calculate an event function at an arbitrary point, specify the point as a starting point for a FORM/SORM analysis. Parameter study, as well as print and display commands are available also for deterministic analyses.

2.10 Parameter Study

It is often desirable to see the evolution of a result (probability, reliability index, crossing rate, moment, sensitivity, function value) over time or as function of any parameter in the model. In Proban this is accomplished by use of the *parameter study* facility.

A parameter study can be assigned to any fixed variable or to any parameter in a distribution or argument in a function that has a numerical value. A number of values are specified. When the parameter study is used, an analysis will be done for each of the specified values.

The main results and importance factors may be presented as a function of the parameter. Each of the individual analyses may also be examined independently.

The following main results may be printed and displayed/plotted as a function of the parameter:

Main results for FORM/SORM, not including equality event(s) or bounds

Prob-FORM	(First Passage) Probability calculated by FORM
Beta-FORM	Reliability (against first passage) index calculated by FORM
Log10P-FORM	Log10 (first passage) probability calculated by FORM
Prob-SORM	Probability calculated by SORM
Beta-SORM	Reliability index calculated by SORM
Log10P-SORM	Log10 of probability calculated by SORM
Crossing-rate-FORM	Crossing rate calculated by FORM

Main results for FORM/SORM using bounds but not including equality event(s)

Prob-Lower	Lower bound of probability
Prob-Upper	Upper bound of probability
Beta-Lower	Lower bound of reliability index
Beta-Upper	Upper bound of reliability index

Prob-Lower	Lower bound of log10(probability)
Prob-Upper	Upper bound of log10(probability)

Main results for FORM/SORM, including equality event(s) and possibly a measured value

dProb-FORM	Derivative of probability
dProb-SORM	Derivative of probability (SORM only)

Main results for Monte Carlo and Directional simulation of a probability

Probability	The probability estimate
Stdv-Prob	Estimated standard deviation of Probability
CoV-Prob	Coefficient of variation for Probability
Conf-Prob-Lo	Lower confidence bound for Probability
Conf-Prob-Up	Upper confidence bound for Probability
Beta	Reliability index corresponding to Probability
Conf-Beta-Lo	Lower confidence bound for Beta
Conf-Beta-Up	Upper confidence bound for Beta
Log10P	Log10(Probability)
Conf-LogP-Lo	Lower confidence bound for Log10P
Conf-LogP-Up	Upper confidence bound for Log10P

Main results for Axis orthogonal simulation of a probability

Probability	The probability estimate
Conf-Prob-Lo	Lower confidence bound for Probability
Conf-Prob-Up	Upper confidence bound for Probability
Beta	Reliability index corresponding to Probability
Conf-Beta-Lo	Lower confidence bound for Beta
Conf-Beta-Up	Upper confidence bound for Beta
Log10P	Log10(Probability)
Conf-LogP-Lo	Lower confidence bound for Log10P
Conf-LogP-Up	Upper confidence bound for Log10P

Correction	The estimated correction to the FORM probability
Stdv-Corr	Estimated standard deviation of Correction
CoV-Corr	Coefficient of variation for Correction
Conf-Corr-Lo	Lower confidence bound for Correction
Conf-Corr-Up	Upper confidence bound for Correction
Prob-FORM	Probability calculated by FORM
Beta-FORM	Reliability index calculated by FORM
Log10P-FORM	Log10(Prob-FORM)

Main results for Monte Carlo and Latin hypercube simulation of a distribution

Mean	The sample mean
Conf-Mean-Lo	Lower confidence bound for Mean
Conf-Mean-Up	Upper confidence bound for Mean
Standard-Dev	The standard deviation of the sample
Skewness	The skewness of the sample
Kurtosis	The kurtosis of the sample

Main results for Deterministic analysis

Value	of <event name> or of <variable name>

2.11 Presentation of Results

During the analysis, Proban displays a short history and summary of the analysis. After an analysis is completed, the results are stored in the database. The results may then be printed and/or displayed at will. Print files and plot files may be generated. Proban provides different levels of print, reaching from a very short summary to a complete listing of all relevant results. In addition, many different plots are possible.

For (first passage) probability analysis, importance factors may be displayed as pie charts.

For distribution analysis, the estimated (Mean value based FORM) or simulated distribution may be displayed together with any other distribution. A simulated distribution may be presented as a histogram or as a cumulative distribution.

After a parameter study has been completed, the main results and the importance factors may be presented as function of the parameter. The same values can be printed in a table. In addition, the individual results can be examined one by one.

3 USER'S GUIDE TO PROBAN

This chapter describes the usage of Proban, with illustrating examples.

Chapter 3 is divided into three parts:

- Section 3.1 to Section 3.8 go through the modelling, analysis and presentation of results, using the different methods available in Proban.
- Section 3.9 and Section 3.10 treat distributions and functions, listing those available in Proban, and describing how to extend the list.
- Section 3.11 contains various hints
- For a quick introduction, read Section 3.1 first.

3.1 How to Do an Analysis

A Proban analysis typically consists of the following steps:

1 Define the model and the questions that are to be answered by Proban.

2 If necessary, program the model function(s) and link with Proban (see Section 3.10.3).

3 Enter the model in Proban and verify it. The commands that are of main interest here are: CREATE, COPY, CHANGE, DELETE, ASSIGN, PRINT, DISPLAY, PLOT, RUN DETERMINISTIC-ANALYSIS and RUN INPUT-CHECK.

4 Set up the analysis and run it. This typically requires usage of DEFINE, SELECT ANALYSIS-METHOD, RUN and possibly SAVE.

5 If the summary results presented during the analysis are not sufficient, examine the results using PRINT RESULT, DISPLAY RESULT, PLOT and possibly also DEFINE RESULT-OPTION and SET.

This process will be illustrated using the following examples.

Example 3.1 A System Network

Consider a simple system network with three components connected in series, and with the first components set up with two redundant spares.

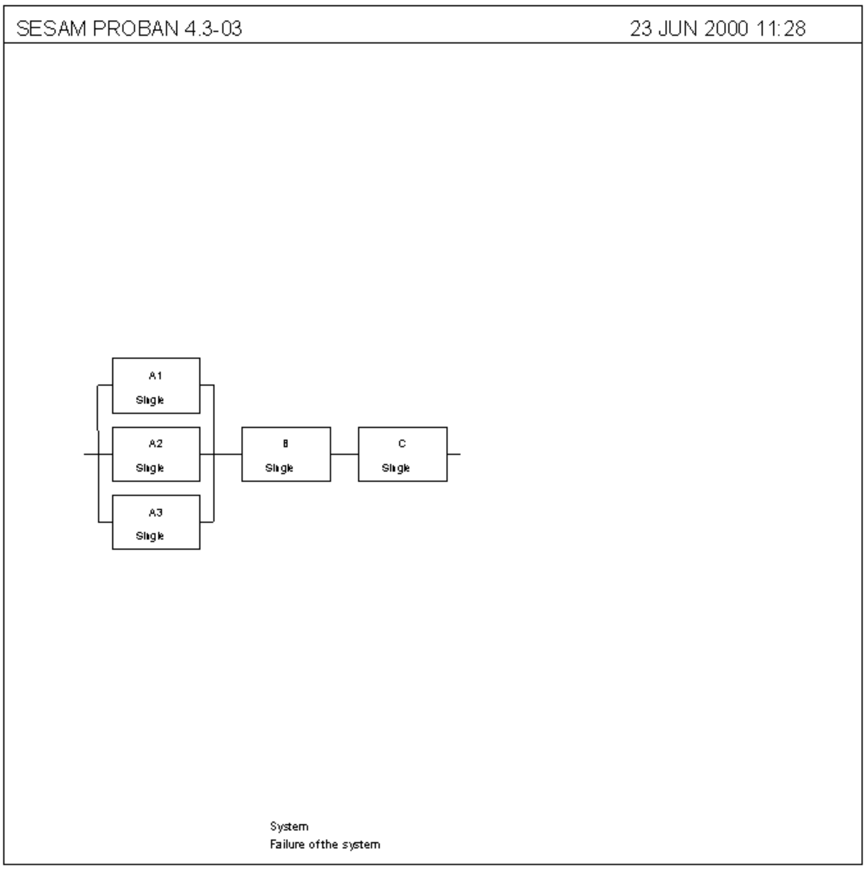


Figure 3.1 A system network

Each component is subjected to a load and has a built-in resistance, and the component fails if the load is greater than the resistance. The load is the same on all components, but their resistance are different.

The distributions of the load and the resistance are:

Table 3.1 Network - Variables

Variable	Type	Parameter/Value
Load	Inv-Gauss distribution	Mean = 80, Stdv = 10, Lower = 0
Resistance of A1,A2,A3	Inv-Gauss distribution	Mean = 110, CoV = 0.1, Lower = 0

Table 3.1 Network - Variables

Resistance of B	Normal distribution	Mean = 120, CoV = 0.1
Resistance of C	Normal distribution	Mean = 130, CoV = 0.1

There are three questions that must be answered:

1 What is the probability that the system will fail?

2 What is the probability that the system will fail if the redundancy inside the A component is removed?

3 Is there a significant effect of replacing any of the components with a component that has a better production quality (i.e. a smaller standard deviation on the resistance)?

Inside Proban the failure criterion on the components can be modelled as differences between the load and the resistance. There is no need to program a model function, as the difference function is already available in Proban.

The following variables are needed in the Proban model:

```
CREATE VARIABLE
    LOOP
    Load 'The common load' DISTRIBUTION Inv-Gauss Mean-StD-Low 80.0 10.0 0.0
    RA1 'Resistance of A' DISTRIBUTION Inv-Gauss Mean-CoV-Low 110 0.1 0.0
    RB  'Resistance of B' DISTRIBUTION Normal    Mean-CoV     120 0.1
    RC  'Resistance of C' DISTRIBUTION Normal    Mean-CoV     130 0.1
    END
COPY VARIABLE RA1 RA2
COPY VARIABLE RA1 RA3
CREATE VARIABLE
    LOOP
    A1 'Failure criterion for component A1' FUNCTION Difference RA1 Load
    A2 'Failure criterion for component A2' FUNCTION Difference RA2 Load
    A3 'Failure criterion for component A3' FUNCTION Difference RA3 Load
    B  'Failure criterion for component B'  FUNCTION Difference RB  Load
    C  'Failure criterion for component C'  FUNCTION Difference RC  Load
    END
```

The event of failure for each component is modelled as a single event: Resistance - Load < 0.

The event of failure of all three A components is modelled as an intersection of the three subevents that the individual A components fail.

The whole system fails if either B1 fails or C1 fails or all three A components fail. This can be modelled as a union of these three subevents. The system with no redundancy in A is modelled similarly.

This is entered into Proban with the following commands:

```
CREATE EVENT
    LOOP
    A1     'Failure of component A1'         SINGLE       A1 < 0.0
    A2     'Failure of component A2'         SINGLE       A2 < 0.0
```

```
A3     'Failure of component A3'         SINGLE       A3 < 0.0
B      'Failure of component B'          SINGLE       B  < 0.0
C      'Failure of component C'          SINGLE       C  < 0.0
A      'Failure of all A components'     INTERSECTION A*
System 'Failure of the system'           UNION        ( ONLY A  B C )
Simple 'Failure with no redundancy in A' UNION        ( ONLY A1 B C )
END
```

The analysis is treated in the following sections. However, the following commands may be used in an interactive session to create the necessary results and print them. Note that the summary results given with the runs will answer questions a) and b).

```
RUN PROBABILITY-ANALYSIS System
PRINT RESULT IMPORTANCE-FACTORS
RUN PROBABILITY-ANALYSIS Simple
```

These commands will run the currently selected probability analysis, which by default is a FORM analysis. Another analysis method may be selected by use of the SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS command.

Example 3.2 Economical Investment

A small company is offered a used computer at the cost of NOK 100000. The company owner estimates that he can have income from selling computer time the following two years of respectively NOK 75000 and NOK 50000. He further estimates that the maintenance costs will be respectively NOK 5000 and 10000 in the two years, and assumes that after two years it will be difficult to sell computer time. It is estimated that the computer then will be sold at NOK 10000. The customers will run the computer themselves via terminals, so the company's cost of running the computer is negligible. There is no other use of the computer. Inflation is assumed to be negligible. The company requires a minimum 10 percent rate of return on its investments.

The Net Present Value can be expressed as:

$$NPV = -C_0 + (I_1 - E_1)/(1+r) + (I_2 - E_2 \div S)/(1+r)^2$$

where C_0 is the initial investment, I_i is the income in year i, E_i is the expense in year i, S is the scrap value and r is the required rate of return.

The variables are assigned the following distributions:

Table 3.2 Example NPV - Variables

Variable	Type	Parameter/Value
C0	Fixed	100000
r	Fixed	0.1
I1	Triangle distribution	Lower = 60000, Mean = 75000, Upper = 90000
I2	Triangle distribution	Lower = 30000, Mean = 50000, Upper = 70000

Table 3.2 Example NPV - Variables

E1	Lognormal distribution	Mean = 5000, Stdv = 1000, Low=0
E2	Lognormal distribution	Mean = 10000, Stdv = 2000, Low=0
S	Normal distribution	Mean = 10000, Stdv = 2000

The expenses are assumed to be positively correlated, with a correlation coefficient of 0.75.

The questions the manager of the company wish to pose, are:

1 What is the distribution of the *NPV*?

2 What is the probability of a loss (i.e. $NPV < 0$)

3 What can be done to increase the profit and/or reduce the risk?

The model function can be modelled directly, using the built in functions, as shown here. It may also be programmed into Proban. The model can be entered into Proban as follows:

```
CREATE VARIABLE
    LOOP
    r 'Required rate of return'FIXED 0.1
    C0 'Initial investment' FIXED 100000
    I1 'Income first year' DISTRIBUTION Triangle Low-Mean-Up 60000 75000 90000
    I2 'Income second year' DISTRIBUTION Triangle Low-Mean-Up 30000 50000 70000
    E1 'Expense first year' DISTRIBUTION Lognormal Mean-StD-Low 5000 1000 0
    E2 'Expense second year' DISTRIBUTION Lognormal Mean-StD-Low 10000 2000 0
    S 'Scrap value' DISTRIBUTION Normal Mean-StD 10000 2000
    END
ASSIGN CORRELATION ( ONLY E1 E2 ) BASIC 0.75

CREATE VARIABLE
    LOOP
    IE1  'I1 - E1'            FUNCTION Difference I1 E1
    r1   '1+r'                FUNCTION Sum ( ONLY 1.0 r )
    Y1   'Year 1'             FUNCTION Division IE1 r1
    IES2 'I2 - E2 + S'        FUNCTION Linear-Comb ( ONLY 1 I2 -1 E2 1 S )
    r2   '(1+r)^2'            FUNCTION Square r1
    Y2   'Year 2'             FUNCTION Division IES2 r2
    NPV  'Net Present Value'  FUNCTION Linear-Comb ( ONLY -1 C0 1 Y1 1 Y2 )
    END
```

Note that expression: $I2 - E2 + S$ is conveniently modelled as a linear combination using the Linear-Comb function with arguments *1, I2, -1, E2, 1, S*: $I2 - E2 + S = 1.0*I2 + -1.0*E2 + 1.0*S$

Note also that the inclusion of a function in Proban with the following syntax: $f(a,r,n) = a /(1+r)$n would ease the modelling in this case. As a further benefit, this function would most likely be reusable in other economical models.

The example illustrates how complex functions can be built through variables referencing variables. However, the same formula can be created using the function formula facility:

```
CREATE FUNCTION NPV 'Net present value' FORMULA
                 ( ONLY r  'Required rate of return'
                        C0 'Initial investment'
                        I1 'Income first year'
                        I2 'Income second year'
                        E1 'Expense first year'
                        E2 'Expense second year'
                        S  'Scrap value'
                 )
                   '-C0+(I1-E1)/(1+R)+(I2-E2+S)/(1+R)**2'
```

A set of formulas can be kept on a journal file and be read into the program whenever needed.

The questions will be answered in the following sections. However, the following commands can be used to get the information needed to answers the questions.

```
ASSIGN SENSITIVITY-CALCULATION VARIABLE *-Mean
RUN DISTRIBUTION-ANALYSIS NPV
SET GRAPH HISTOGRAM FILLING HOLLOW
DISPLAY DISTRIBUTION ( Empirical *Fit )
    LOOP
    DENSITY
    DISTRIBUTION
    END
PRINT RESULT ALL
RUN PROBABILITY-ANALYSIS SINGLE-EVENT NPV < 0
```

Proban can also be used to do a probability or distribution analysis, conditional on some obtained information.

Suppose that the manager decides to go for the project. After 6 months, he is certain that the income after the first year will exceed 70000. This information can be used to update the distribution of the NPV and the probability of a loss.

This can be formulated as a conditional probability:

$$P(NPV < 0 \mid I_1 > 70000) \;=\; P(NPV < 0 \cap I_1 > 70000)/P(I_1 > 70000)$$

The following commands will recalculate the values, conditioned on the new information:

```
RUN DISTRIBUTION-ANALYSIS CONDITIONED NPV SINGLE-EVENT I1 > 70000
RUN PROBABILITY-ANALYSIS CONDITIONED SINGLE-EVENT NPV < 0 SINGLE-EVENT I1 > 70000
```

Result presentation is as above.

3.2 Presentation of Model Data and Results

Results and input data are presented using the PRINT, DISPLAY and PLOT commands. The SET command may be used to control print and display options and to control the output print and plot file(s). The DEFINE RESULT-OPTION command is used to set some options specific to the presentation of a selected results.

When many results are stored simultaneously, only the currently selected result can be presented. Use SELECT RESULT to access a particular result. After an analysis is completed, the result is named: “LastAnalysis” and becomes the selected result. It is necessary to save the result under another name (using SAVE RESULT) if it is not to be over written in the next run.

3.2.1 Print

The PRINT command is used to present data in tabular formats.

By default, the print is sent to the screen - that is the terminal (window) when running in line mode or a separate print window when running in graphics mode. This destination is always effective when the program starts, even if the setting was changed in a previous session using the same database. The print may also be directed to a file. The print destination and print file name is controlled by use of the SET PRINT command. The default file name is identical to the database and journal file name. The print file always has the extension “.lis”.

The printed output on a file will be slightly different from the screen print. A page header is added and in some cases also a nomenclature.

On printing to screen in an interactive line mode session, Proban will prompt at the end of each page for an action. At this prompt it is possible to abort the print, or to browse through the previous print, or to continue printing. These prompts are not issued when running graphics mode. Instead, the print window has a scroll bar that may be used to examine the print after it has been presented.

The number of lines in a screen page may be set using the SET PRINT command (see also the description there). This number controls when prompts are issued (as described above) as well as the insertion of intermediate headers in a table when the table scrolls out of the screen.

Some of the print tables used for model verification are shown below.

The following table shows the print of a distribution variable:

```
PRINT VARIABLE RA1

                              +------------------+
                              !    Variable      !
                              +------------------+
                              !       RA1        !
                              ! Resistance of A  !
                              +------------------+

Type         Name            Dim Parameter    Value                     Sens
----------------------------------------------------------------------------
Distribution Inv-Gauss         1 Mean         110.0                     Off
                                 Coef-of-Var  0.1                       Off
                                 Lower-Bound  0.0                       Off

Calculated parameters:           Stand-Dev    11.0

                                 Skewness     0.3
                                 Kurtosis     3.15
                                 Median       109.453185
```

Note that those moments, that are not given as input, will be calculated and printed when this is possible.

This is the print of a function variable:

```
PRINT VARIABLE A1

                    +-------------------------------------+
                    !               Variable              !
                    +-------------------------------------+
                    !                  A1                 !
                    ! Failure criterion for component A1 !
                    +-------------------------------------+

Type         Name             Dim Parameter    Value                    Sens
-----------------------------------------------------------------------------
Function     Difference         1 Additive-Arg RA1                      Off
                                  Subtract-Arg Load                     Off
```

This it the print of the system event in Example 3.1:

```
PRINT EVENT System

                           +------------------------+
                           !         Event          !
                           +------------------------+
                           !         System         !
                           ! Failure of the system !
                           +------------------------+

Event-type   Subevent     Subtype      Contents
--------------------------------------------------------------------------------
Union        A            Intersection 3 sub-events
             B            Single       B < 0.0
             C            Single       C < 0.0
```

Other event types are printed similarly.

The correlation between the two expenses in Example 3.2 is printed with the command:

```
PRINT CORRELATION ( INCLUDE * )
```

Because it is the only correlation that has been defined, no other correlation will be printed. The result is:

```
                      +---------------------------------+
                      ! Correlations between variables !
                      +---------------------------------+

           Variable 1   Variable 2   Input         Basic  Normalized
           ------------------------------------------------------
           E1           E2           Basic         0.7500  0.7537
```

Note that the normalised correlation is also printed.

3.2.2 Display and Plot

The DISPLAY command is used to view data graphically. The PLOT command is used to send the last display to a file.

Displays are by default sent to the screen. The operation of such a display window depends on the device used. The display device is set by use of the SET DISPLAY DEVICE command.

If a display to screen is attempted to an incorrect device, the terminal (window) will most likely be filled with strange characters, and it may be necessary to issue a few <Return>s in order to get back to the main prompt.

A display may also be directed to a file. The display destination and the plot file name are controlled by the commands SET DISPLAY DESTINATION and SET PLOT FILE respectively.

The last display may be sent to the current plot file (even if the current display destination is to the screen) by issuing the PLOT command. This command will actually process the last display again and send it to the file - it will not just take a copy of the previous display. This implies that if any display settings or other input has been changed, the plot file version may be different from the display that was shown on the screen.

Several plot file formats are available. See the description of the SET PLOT command.

It is not possible to write plots with different colour options to the same file.

The DISPLAY command remembers the last command it executed and presents it as default the next time DISPLAY is entered. Thus the command: DISPLAY ; (semicolon) is a simple way of repeating the previous display command.

To display the input distributions in Example 3.1, and take a copy on a file, use the following commands:

```
DISPLAY DISTRIBUTION ( ONLY Load RA1 RB RC )
    LOOP
    DISTRIBUTION
    # PLOT
    DENSITY
    # PLOT
    END
```

Note the inserted plot commands, that are executed without leaving the loop.

Proban may also be used to display events. The plot in Figure 3.1 was generated by using the command: DISPLAY EVENT System MULTIPLE.

The two distribution plots looks like this (with the display frame off):

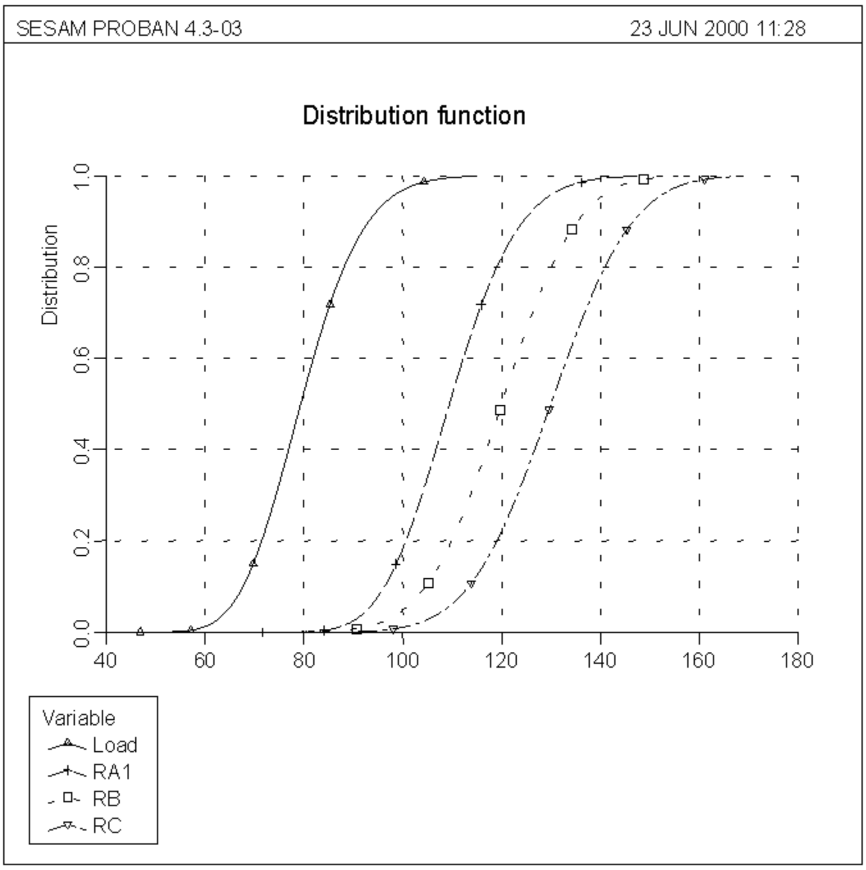


Figure 3.2 Distribution functions for variables in Example 3.1

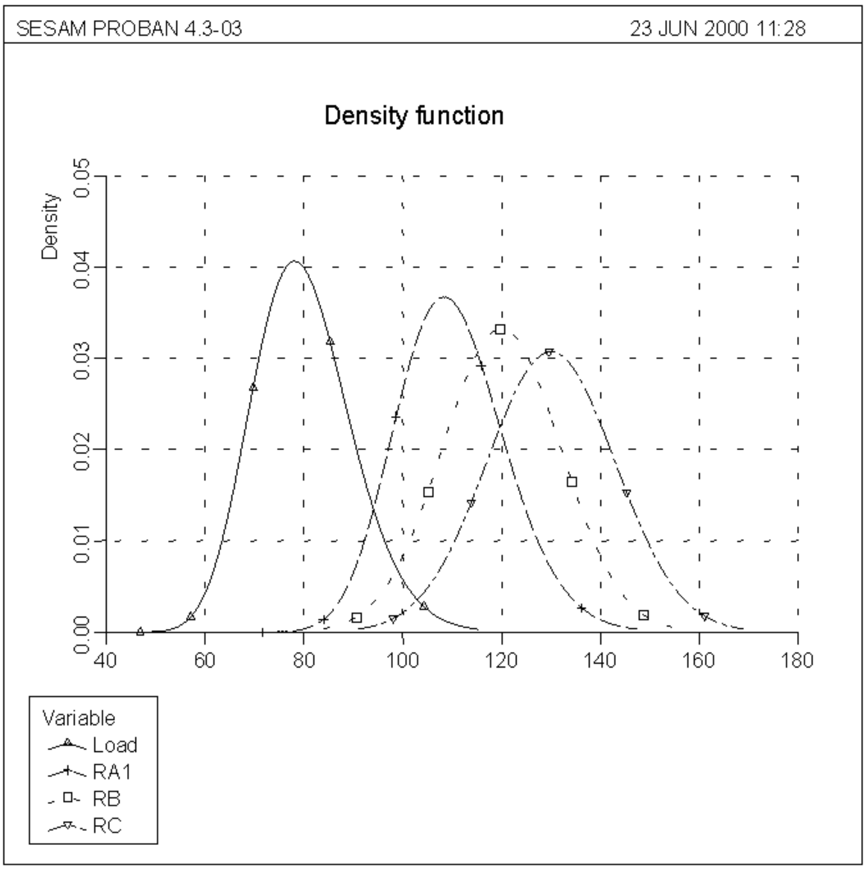


Figure 3.3 Density functions for variables in Example 3.1

3.3 Probability Analysis and Results

Performing a probability analysis requires the following steps, after the model has been specified:

1 Select the method to be used for probability analysis, using SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS. The default method is FORM when Proban starts from a new database.

2 Define the desired options for the chosen method and/or general analysis options. These options are explained in the DEFINE command. The default options will be sufficient in most cases.

3 ASSIGN SENSITIVITY-CALCULATION to the required parameters and/or decide the extent of sensitivity calculation using DEFINE ANALYSIS-OPTION SENSITIVITY.

4 Decide if importance factors are to be calculated by use of the DEFINE ANALYSIS-OPTION IMPORTANCE-FACTORS.

5 Run the analysis using RUN PROBABILITY-ANALYSIS.

6 Present the results using PRINT RESULT, DISPLAY RESULT and PLOT.

The different analysis methods are described in separate sections, using the examples from Section 3.1. FORM and SORM are treated together because of their similarity.

3.3.1 FORM/SORM

First a word of caution: FORM requires the model function(s) and distribution function(s) to be differentiable, and SORM requires them also to be twice differentiable at the design point. If they are not, the results will be unreliable when the design point is close to a point with a lack of differentiability. The model in Example 3.2 exemplifies this. The triangle distribution has a density function that is not differentiable at its most likely argument (in the middle of the distribution). This causes the SORM result to be unreliable in the middle of the distribution, and the importance factors to have a strange behaviour at the same area. See also the end of Section 3.8.

Consider Example 3.1, described in Section 3.1. An analysis of the probability of failure of component A with the default settings is done with the following command:

```
RUN PROBABILITY-ANALYSIS A
```

and produces the following messages while the analysis is running.

```
      Starting Probability Analysis of: A

      Starting FORM calculation

      Starting linearization of:
      Intersection of:  A1  A2  A3
      Linearization completed
      Calculating importance factors

      FORM  Reliability index:       2.8249
      FORM  Probability:        2.36457E-03
```

The following commands can be used to see the results:

```
PRINT RESULT ALL
PRINT RESULT ANALYSIS-SETTINGS
PRINT RESULT SUMMARY
PRINT RESULT IMPORTANCE-FACTORS
```

There is also a command that is used to print sensitivity results (PRINT RESULT SENSITIVITY). This command is not available after this analysis, because no parametric sensitivity values were calculated.

The summary print produces the following output:

```
PRINT RESULT SUMMARY
```

```
+-----------------------------------------------+
! Probability of : A                             !
!                  Failure of all A components !
! Analysis method: FORM                          !
+-----------------------------------------------+


  FORM Probability:                 2.36457E-03

  FORM Reliability index:                2.8249


     U-space Geometry:   Small Intersection

       Number of linearization points:   1


            Description of subevents

Subevent     Definition
-----------------------------------------------------
A1           A1 < 0.0
A2           A2 < 0.0
A3           A3 < 0.0
```

The importance factor print is as follows:

```
PRINT RESULT IMPORTANCE-FACTORS
```

```
+-----------------------------------------------+
! Probability of : A                             !
!                  Failure of all A components !
! Analysis method: FORM                          !
+-----------------------------------------------+



               Importance factors

            Variable     Importance
            -----------------------
            Load               82.0
            RA2                 6.0
            RA3                 6.0
            RA1                 6.0
```

It lists the importance factors in order of magnitude. If there are many small importance values, they can be cut off from the print by use of the command: DEFINE RESULT-OPTION IMPORTANCE-CUTOFF.

The linearisation point is printed as part of the output from the PRINT RESULT ALL command. It looks like this:

```
                Linearization point number 1 of A

      Subevent      Definition
      ------------  ----------------------------------------
      A1           A1 < 0.0
      A2           A2 < 0.0
      A3           A3 < 0.0

     Variable     Type               Value        Prob
     --------------------------------------------------------
     RA1          Inv-Gauss     1.033881161E+02 0.284038  -
     Load         Inv-Gauss     1.033881089E+02 0.983141  ++
     RA2          Inv-Gauss     1.033881161E+02 0.284038  -
     RA3          Inv-Gauss     1.033881161E+02 0.284038  -
     A1           Difference    7.103621499E-06
     A2           Difference    7.103621499E-06
     A3           Difference    7.103621499E-06
```

The + and - indications at the right show if a variable has a load effect (+) or a resistance effect (-). The number of + or - indicates the strength of the effect. The linearisation point has been transformed back to the input space of the variables, even though the linearisation actually took place in U-space. The values of those variables, that are not distributions, are also shown. The probability column shows the probability corresponding to the fractile at the linearisation point for each distribution variable (e.g. 103.39 is the 98.3% fractile in the distribution for the load). The V-space point may also be added to this table by use of the command: DEFINE RESULT-OPTION V-SPACE-POINT. The V-space point is the fractile in the standard normal distribution that corresponds to the probability value.

A SORM analysis of the system without redundancy in component A, and including a sensitivity analysis on all parameters, can be done as follows:

```
SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS SORM PARABOLIC
DEFINE ANALYSIS-OPTION SENSITIVITY ALL
RUN PROBABILITY-ANALYSIS Simple
```

The message appearing while the analysis is running is:

```
Starting Probability Analysis of: Simple

Starting SORM calculation

Starting linearization of:
Union of:  A1  B  C
Linearization completed
Calculating importance factors and 10 parametric sensitivity values

SORM  Reliability index:       1.9131
SORM  Probability:        2.78657E-02
```

Note that this is a large intersection, It will therefore produce one linearisation point for each subevents (component). The print of the linearisations (PRINT RESULT ALL) is not shown here. Note also that Proban does not simultaneously provide FORM and SORM results for a large intersection. It does so for any other geometry.

Note that Proban informs about how many parametric sensitivity values it calculates. The sensitivity results are presented in three tables, one for the reliability index, one for the probability and one for the logarithm of the probability. The tables are similar. In this case, the table for the reliability index is:

```
             Parametric sensitivity result for Beta = 1.9131294288

   Variable     Type         Parameter      Value    dBeta/dPar  Measure
   ----------------------------------------------------------------------
   RA1          Inv-Gauss    Mean          1.100E+02  4.384E-02   0.48220
                             Coef-of-Var   1.000E-01 -6.803E+00  -0.06803
                             Lower-Bound   0.000E+00  2.354E-04 Undefined
   Load         Inv-Gauss    Mean          8.000E+01 -6.555E-02  -0.52437
                             Stand-Dev     1.000E+01 -1.112E-01  -0.11124
                             Lower-Bound   0.000E+00 -5.525E-04 Undefined
   RB           Normal       Mean          1.200E+02  1.036E-02   0.12429
                             Coef-of-Var   1.000E-01 -3.025E+00  -0.03025
   RC           Normal       Mean          1.300E+02  2.241E-03   0.02913
                             Coef-of-Var   1.000E-01 -9.421E-01  -0.00942
```

The table lists the variable name and type, and each parameter name and value. Then follows the sensitivity value dBeta/dPar and finally, when possible, the sensitivity measure. The measure is defined in Section 2.8. In this case, it shows the estimated change in β given a 10% increase in the parameter. The sensitivity measure shows at a glance, that the mean of the Load and the mean of RA1 are the two most important parameters.

The analysis of the total system using SORM without parametric sensitivity analysis is produced as follows:

```
DEFINE ANALYSIS-OPTION SENSITIVITY NONE
RUN PROBABILITY-ANALYSIS System

      Starting Probability Analysis of: System

      Starting SORM calculation

      Starting linearization of:
      Intersection of:  A1  A2  A3
      Linearization completed
      Calculating importance factors

      FORM  Reliability index:       2.8249
      SORM  Reliability index:       2.8256
      FORM  Probability:        2.36457E-03
      SORM  Probability:        2.35961E-03

      Starting linearization of:
      Single event:  B
      Linearization completed
```

```
Calculating importance factors

FORM  Reliability index:       2.4915
SORM  Reliability index:       2.4699
FORM  Probability:        6.36053E-03
SORM  Probability:        6.75823E-03

Starting linearization of:
Single event:  C
Linearization completed
Calculating importance factors

FORM  Reliability index:       2.9483
SORM  Reliability index:       2.9262
FORM  Probability:        1.59742E-03
SORM  Probability:        1.71569E-03


Lower bound on Reliability index:       2.3261
Upper bound on Reliability index:       2.3347
Lower bound on Probability:        9.77831E-03
Upper bound on Probability:        1.00065E-02
```

This analysis provides bounds instead of a direct probability, because of the geometry of the limit state surface in U-space. The system event is a union of events, with at least one intersection between the subevents. The only way Proban can treat this using FORM/SORM is to analyse each subevent by itself, and then use these results to bound the probability of the union event.

A union of single events may also be analysed using the bounding technique. Use the command DEFINE FORM/SORM BOUNDS ON to achieve this.

The first questions in Example 3.1 can now be answered. Using SORM, the probability of failure is about 0.01, corresponding to a reliability index of about 2.33. The effect of removing the redundancy in component A is to increase the failure probability to about 0.028, corresponding to a reliability index of about 1.9.

A print of the summary results and of all results yield, in addition to the bounds and the summary result for each of the subevents, a list of intersection probabilities:

```
                    Subevent Intersection Probabilities:

    Subi Subj Probability  Subi Subj Probability  Subi Subj Probability
    ---------------------------------------------------------------------
       1    1  2.35961E-03
       2    1  5.69582E-04    2    2  6.75823E-03
       3    1  2.28220E-04    3    2  2.57420E-04    3    3  1.71569E-03
```

The intersection probabilities are used to calculate the probability bounds. They are probabilities of the intersections between pairs of the subevents.

The importance factors are the key to the last question of Example 3.1 (regarding the importance of the uncertainty of the resistance values). A print of the importance factors produces, because this is a bounds analysis, a table for the main event and a table for each subevent. The main table is shown here:

```
+----------------------------------------+
! Probability of : System                !
!                  Failure of the system !
! Analysis method: FORM                  !
+----------------------------------------+


            Importance factors

         Variable     Importance
         -----------------------
         Load               74.0
         RB                 22.8
         RC                  2.0
         RA3                 0.4
         RA1                 0.4
         RA2                 0.4
```

The importance factors reveal that it would pay to reduce the uncertainty on the resistance of RB, and that the uncertainty in the other resistance is insignificant. However, the really significant contribution comes from the uncertainty in the load.

Using the estimate in Section 2.8, it can be predicted that if the standard deviation on RB is removed, the lower bound reliability index will change to $2.3261/\sqrt{(1 - 0.228)} = 2.6474$ while the upper bound reliability index will change to 2.6572. If RB is changed to a fixed variable with value 120, and the analysis is run again, the result gives the following bounds on the reliability index: $2.6541 \le \beta \le 2.6645$, so in this case the prediction was quite good.

When analysing equality events (i.e. an event of the type: B = 0), the results are different. Such events do not generate probabilities. The probability of an equality event is always zero when the distributions are continuous. Instead, what is calculated is the derivative of the probability with respect to the right hand side of the equality event(s). In case the analysis is of a single equality event, the result is the value of the density function for the random variable that is used to define the event, calculated at the right hand side threshold value for the event.

Equality events cannot be used in analyses that require calculation of bounds, involves a probability variable or a time dependent stochastic process.

Equality events mostly come up in analyses involving inspection and updating, where a quantity is observed to be equal to some value. In this situation, however, the variable which keeps the measurement information must be assigned to the equality event since the corresponding conditional probability is calculated from sensitivity factors with respect to the measured value. The two examples: Fatigue crack growth and Creep in concrete from the example manual /3/ contain analyses of inspection and updating using equality events.

However, in order to document the results, consider the follow hypothetical example:

```
CHANGE EVENT B : SINGLE B = 0
```

```
RUN PROBABILITY-ANALYSIS Simple

      Starting Probability Analysis of: Simple

      Starting FORM calculation

      Starting linearization of:
      Union of:  A1  B  C
      Linearization completed
      Calculating importance factors

      FORM  Derivative of Probability:  1.48788E-05
```

The print is similar to the print, that has been described previously.

Conditional probability calculation is straightforward to execute. This will be demonstrated using Example 3.2.

To calculate the probability:

$$P(NPV < 0 \mid I_1 > 70000) = P(NPV < 0 \cap I_1 > 70000) / P(I_1 > 70000)$$

using FORM/SORM, Proban first finds the intersection probability in the numerator, then the probability in the denominator, and finally divides to get the conditional probability.

Importance factors and parametric sensitivity values may be calculated with a conditional probability. In this case the importance factors and the sensitivity values with respect to the mean values is calculated.

```
ASSIGN SENSITIVITY-CALCULATION VARIABLE ONLY *-Mean*
      Assigned sensitivity calculation to the Mean of E1
      Assigned sensitivity calculation to the Mean of E2
      Assigned sensitivity calculation to the Mean of I1
      Assigned sensitivity calculation to the Mean of I2
      Assigned sensitivity calculation to the Mean of S

RUN PROBABILITY-ANALYSIS CONDITIONED SINGLE-EVENT NPV < 0
                                     SINGLE-EVENT I1  > 70000


      Starting Probability Analysis of: NPV < 0.0 given I1 > 70000.0

    * The Triangle density for variable I1 is not differentiable everywhere.
    * The Triangle density for variable I2 is not differentiable everywhere.

    * WARNING  The model does not fulfil the differentiability requirements of the
      selected analysis. As a consequence the analysis may not work, or the
      results (particularly sensitivities) may be wrong

      Analysing intersection event in conditional calculation

      Starting SORM calculation
```

```
Starting linearization of:
Intersection of:  NPV < 0.0  I1 > 70000.0
Linearization completed
Calculating importance factors
Calculating 5 parametric sensitivity values

FORM Reliability index:        1.0238
SORM Reliability index:        0.9621
FORM Probability:         1.52965E-01
SORM Probability:         1.67999E-01

Analysing conditioning event in conditional calculation

Starting SORM calculation

Starting linearization of:
Single event:  I1 > 70000.0
Linearization completed
Calculating importance factors and 5 parametric sensitivity values

FORM  Reliability index:      -0.7647
SORM  Reliability index:      -0.7647
FORM  Probability:        7.777786-01
SORM  Probability:        7.777786-01


Final results: NPV < 0.0 given I1 > 70000.0

SORM  Reliability index:       0.7858
SORM  Probability:        2.15999E-01
```

Only the conditional result itself is printed. The results from the intersection event and the conditioning event are not available. If they are of interest, the corresponding models must be defined and analysed separately.

3.3.2 Monte Carlo Simulation

Monte Carlo simulation is the simplest way to simulate a probability. The result is unbiased, but it may have a large standard deviation.

It is not possible in Proban to calculate sensitivities or importance factors using Monte Carlo simulation of a probability.

Consider the probability of a loss in Example 3.2. The following commands will simulate this probability (Proban messages are also shown):

```
SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS MONTE-CARLO-SIMULATION
RUN PROBABILITY-ANALYSIS SINGLE-EVENT NPV < 0


      Starting Probability Analysis of: NPV < 0.0
```

```
Starting Monte Carlo simulation

Stopping after 1000 simulations or 60.0 CPUsec
Simulations completed: 250
Simulations completed: 500
Simulations completed: 750
Simulations completed: 1000

Number of simulations:               1000
Number in intersection event:         330
Estimated probability:         3.3000E-01
Standard dev. of Probability:  1.4869E-02
Coeff of Var. of Probability:       0.045
Estimated Reliability index:       0.4399
```

The accuracy here was quite good (a coefficient of variation of about 5%). Note that the stop criteria (60 seconds or 1000 simulations) are shown. It is also possible to demand a stop if a required coefficient of variation has been reached. The stop criteria is manipulated using the command: DEFINE PROBABILITY-SIMULATION MONTE-CARLO.

A summary of the results may be printed:

```
PRINT RESULT SUMMARY
```

```
                  +------------------------------------------+
                  ! Probability of : NPV < 0.0               !
                  !                  Net Present Value       !
                  ! Analysis method: Monte Carlo simulation  !
                  +------------------------------------------+


                     Final results after 1000 simulations

                      Estimate  Stand-Dev C-of-V 90% confidence interv
                     --------- ---------- ------ ---------------------
       Probability   3.300E-01  1.487E-02  0.045  3.055E-01  3.545E-01
       Beta          0.4399                       0.3733     0.5085
       Log10(Prob)  -4.815E-01                   -5.149E-01 -4.504E-01
```

This gives confidence intervals in addition to the previous information shown during the run. The confidence level may be set using the command: DEFINE RESULT-OPTION CONFIDENCE-VALUE.

The print option PRINT RESULT ANALYSIS-SETTINGS shows the analysis settings used, among other things the seeds used by the random generator.

```
PRINT RESULT ANALYSIS-SETTINGS
```

```
                  +------------------------------------------+
                  ! Probability of : NPV < 0.0               !
                  !                  Net Present Value       !
                  ! Analysis method: Monte Carlo simulation  !
```

```
                  +------------------------------------------+


Method                  Option                  Value
------------------------------------------------------------------------------
Analysis Method         Probability             Monte Carlo simulation

Monte Carlo Sim (Prob)  Stop Criteria:          Simulations: 1000
                                                CPU seconds: 60.0
                                                Coef of Var: No requirement
Analysis Option         Parameter Study         Off
                        Sensitivity             Selected
                        Seeds                   Seed 1:  216264090
                                                Seed 2: -276250807
                                                Seed 3:  326643946
General info            Time of analysis        00:15:50 10-FEB-1992
                        CPU time used           16 seconds
```

The results may be reproduced exactly by using the same seeds and the same number of simulations. The seeds are manipulated using the command: DEFINE ANALYSIS-OPTION SEEDS. This also applies to all other simulation methods in Proban.

The PRINT RESULT ALL command generates the summary print plus a history of intermediate results during the simulation. This history table is:

```
                       Intermediate simulation results:

 NoSim Probability   StDv(Prob)  C-of-V   Beta       Log10(Prob)
-----------------------------------------------------------------
   100  3.50000E-01  4.79372E-02  0.137   0.38532    -4.55932E-01
   200  3.30000E-01  3.33325E-02  0.101   0.43991    -4.81486E-01
   300  3.06667E-01  2.66667E-02  0.087   0.50532    -5.13333E-01
   400  3.20000E-01  2.33530E-02  0.073   0.46770    -4.94850E-01
   500  3.12000E-01  2.07406E-02  0.066   0.49019    -5.05845E-01
   600  3.23333E-01  1.91117E-02  0.059   0.45840    -4.90350E-01
   700  3.27143E-01  1.77456E-02  0.054   0.44782    -4.85263E-01
   800  3.25000E-01  1.65699E-02  0.051   0.45376    -4.88117E-01
   900  3.31111E-01  1.56958E-02  0.047   0.43685    -4.80026E-01
  1000  3.30000E-01  1.48769E-02  0.045   0.43991    -4.81486E-01
```

The table is useful to check if the simulation has stabilised. This can be seen from the development of the coefficient of variation. It should decrease steadily. If it does not, the result may actually be more inaccurate than it appears, and further simulation is recommended (see RUN RESTART). The number of lines in the table is controlled by using the command: DEFINE RESULT-OPTION INTERMEDIATE-RESULTS.

The simulation may be restarted from the previous result by using the command: RUN RESTART. The stop criteria may be changed before the run is restarted. This is useful e.g. for estimating the time a simulation need to run in order to produce a required accuracy on the result, or for continuing a simulation that did not produce the desired accuracy.

A conditional probability is calculated just like any other probability. In this case Proban counts the number of hits in the conditioning event, and the number of hits in the intersection event. The probability estimate is then the division of these two values.

3.3.3 Directional Simulation

Directional simulation is a more sophisticated version of Monte Carlo simulation. It can be used to simulate sensitivities and importance factors.

Consider again the probability of a loss in Example 3.2. The following commands will simulate this probability including parametric sensitivities for all the mean parameters (the messages given by Proban are also shown):

```
SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS DIRECTIONAL-SIMULATION
ASSIGN SENSITIVITY-CALCULATION VARIABLE ONLY *-Mean*
      Assigned sensitivity calculation to the Mean of E1
      Assigned sensitivity calculation to the Mean of E2
      Assigned sensitivity calculation to the Mean of I1
      Assigned sensitivity calculation to the Mean of I2
      Assigned sensitivity calculation to the Mean of S
RUN PROBABILITY-ANALYSIS SINGLE-EVENT NPV < 0


      Starting Probability Analysis of: NPV < 0.0

      Starting Directional simulation

      Stopping after 50 simulations or 60.0 CPUsec
      Simulating importance factors and 5 sensitivity values
            12 simulations completed.
            24 simulations completed.
            36 simulations completed.
            48 simulations completed.
      CPU time limit exceeded

      Number of simulations:                 50
      Estimated Probability:         2.9244E-01
      Standard dev. of Probability:  9.5408E-03
      Coeff of Var. of Probability:       0.033
      Estimated Reliability index:       0.5463
```

The messages are very similar to the messages produced by the other simulation methods. This time the CPU time limit was the effective stop criterion. It is also possible to demand a stop if a required coefficient of variation has been reached. The stop criteria are manipulated using the command: DEFINE PROBABILITY-SIMULATION DIRECTIONAL. This command is also used to define the search method and the simulation method.

The summary print is identical to the print for Monte Carlo simulation.

The importance factors are printed using the ALL or IMPORTANCE-FACTORS option.

```
PRINT RESULT IMPORTANCE-FACTORS

                    +------------------------------------------+
                    ! Probability of : NPV < 0.0               !
                    !                  Net Present Value       !
                    ! Analysis method: Directional simulation  !
                    +------------------------------------------+


                             Importance factors

                       Variable       Importance  StDv(Imp)
                       ----------------------------------
                       I2                   55.1        4.8
                       I1                   35.4        4.5
                       ImpGroup-1            6.5        1.0
                       S                     3.1        0.6

ImpGroup-1  :  E1           E2
```

Note the importance group, that is created from the two correlated expense variables. When two or more distribution variables are correlated, they will generate only one importance factor together. Note also that standard deviations are given.

This table shows clearly, that if the manager could be more certain about his income from this project, he would reduce his probability of a loss. It is not nearly as important to control the uncertainty on the expenses.

The importance factors may also be displayed in a pie chart. The following commands generate a plot file with the importance factor plot without generating a screen display:

```
SET DISPLAY DESTINATION FILE
DISPLAY RESULT IMPORTANCE-FACTORS
```

The PLOT command could have been used after the display instead of setting the display destination to file.

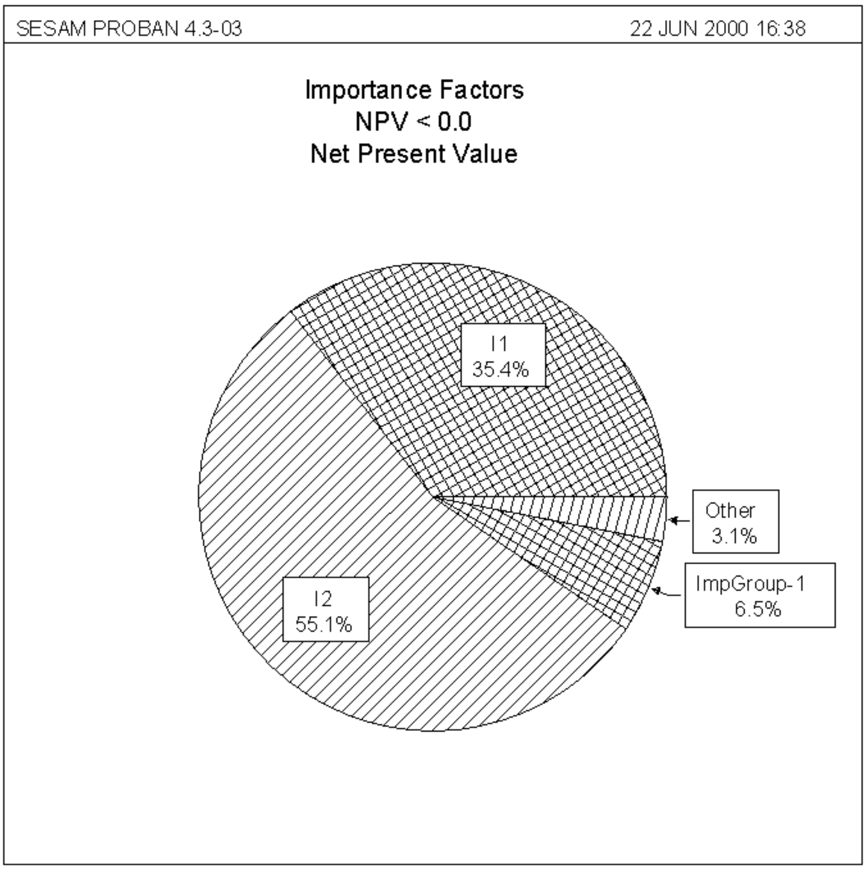


Figure 3.4 Importance factors for probability of loss in Example 3.2

The sensitivity factors are shown in three tables. One for the probability, one for the reliability index and one for the logarithm of the probability. The tables have the same layout. The table for the probability is:

```
         Parametric sensitivity result for Probability = 0.29648506009

Variable      Type          Parameter      Value    dProb/dPar SD(deri)  Measure
--------------------------------------------------------------------------------
I1            Triangle      Mean         7.500E+04 -3.571E-05 2.76E-06 -2.68E-01
E1            Lognormal     Mean         5.000E+03  3.372E-05 2.77E-06  1.69E-02
I2            Triangle      Mean         5.000E+04 -3.291E-05 2.56E-06 -1.65E-01
E2            Lognormal     Mean         1.000E+04  3.059E-05 2.50E-06  3.06E-02
S             Normal        Mean         1.000E+04 -3.060E-05 2.50E-06 -3.06E-02
```

Note that the table includes a standard deviation SD(deri) of dProb/dPar, because the sensitivity value is a simulated value. The other entries are described in Section 3.3.1.

The PRINT RESULT ALL command generates a print of the summary, intermediate simulation results, importance factors and sensitivity results. The intermediate result table is identical to the table presented in the description of Monte Carlo simulation in Section 3.3.2.

It is possible to print and display the sample of probabilities using the commands PRINT RESULT SAMPLE and DISPLAY RESULT DISTRIBUTION. These commands and the results are described in Section 3.6.1.

A conditional probability is calculated just like any other probability. The analysis will be slower, because Proban needs to calculate both the intersection event probability and the conditioning event probability. The resulting probability is a division of the estimates of the intersection probability and the conditioning probability. For this reason, there is no sample of independent and identically distributed conditional probabilities, and therefore the PRINT RESULT SAMPLE command is not in effect in this case.

The simulation may be restarted from the previous result by using the command: RUN RESTART. The stop criteria may be changed before the run is restarted. This is useful e.g. for estimating the time a simulation will run in order to produce a required accuracy on the result, or for continuing a simulation that did not produce a sufficient accuracy.

3.3.4 Axis Orthogonal Simulation

Axis orthogonal simulation is used to estimate a correction to the FORM probability. The correction may be additive or multiplicative, depending on the type of sampling density used.

It is not possible to simulate importance factors or parametric sensitivities by use of axis orthogonal simulation.

Consider again the probability of a loss in Example 3.2. The following commands will simulate this probability using axis orthogonal simulation: (the messages given by Proban are also shown):

```
SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS AXIS-ORTHOGONAL-SIMULATION
RUN PROBABILITY-ANALYSIS SINGLE-EVENT NPV < 0

      Starting Probability Analysis of: NPV < 0.0

      Starting FORM calculation

      Starting linearization of:
      Single event:  NPV < 0.0
      Linearization completed
      Calculating importance factors and 5 parametric sensitivity values

      FORM  Reliability index:       0.5470
      FORM  Probability:        2.92197E-01


      Starting Axis orthogonal simulation

      Stopping after 50 simulations or 60.0 CPUsec
            12 simulations completed.
            24 simulations completed.
            36 simulations completed.
```

```
      48 simulations completed.

Number of simulations:                  50
Estimated Correction :          1.0340E+00
Standard dev. of Correction:    1.0151E-02
Coeff of Var. of Correction:         0.010
Estimated Probability:          3.0213E-01
Standard dev. of Probability:   2.9661E-03
Coeff of Var. of Probability:        0.010
Estimated Reliability index:        0.5183
```

The simulation requires a FORM analysis to run, hence the FORM result. The sensitivity analysis applies to the FORM result, not to the simulation.

The multiplicative correction to the FORM probability is simulated by default. To change to the additive correction, use the command: DEFINE PROBABILITY-SIMULATION AXIS-ORTHOGONAL DENSITY STANDARD-NORMAL. As can be seen, the correction is small in this case.

The stop criteria for the simulation are manipulated using the command: DEFINE PROBABILITY-SIMULATION AXIS-ORTHOGONAL. It is possible to demand a stop if a required coefficient of variation has been reached. This command is also used to define the search method.

The summary print looks like this:

```
PRINT RESULT SUMMARY

                +---------------------------------------------+
                ! Probability of : NPV < 0.0                  !
                !                  Net Present Value          !
                ! Analysis method: Axis Orthogonal simulation !
                +---------------------------------------------+


                      Final results after 50 simulations

                      Estimate  Stand-Dev C-of-V 90% confidence interv
                     --------- ---------- ------ ---------------------
       Correction    1.034E+00  1.015E-02  0.010  1.017E+00  1.051E+00
       Probability   3.021E-01  2.966E-03  0.010  2.973E-01  3.070E-01
       Beta          0.5183                       0.5043     0.5323
       Log10(Prob)  -5.198E-01                   -5.269E-01 -5.128E-01
```

It includes a line showing the results for the simulated correction value. The standard deviation of the probability is derived from the standard deviation of the correction, not from a sample of probabilities. The confidence level (default 90%) may be changed by use of the command: DEFINE RESULT-OPTION CONFIDENCE-VALUE.

PRINT RESULT ALL produces in addition to the summary table a table of intermediate results, showing again the simulated correction:

```
 NoSim  Correction   StDv(Corr)  C-of-V   Beta       Log10(Prob)  Probability
-------------------------------------------------------------------------------
     5  1.03647E+00  1.06515E-02  0.010   0.51621    -5.18768E-01  3.02853E-01
    10  1.03600E+00  7.99191E-03  0.008   0.51661    -5.18965E-01  3.02716E-01
    15  1.03659E+00  6.77490E-03  0.007   0.51611    -5.18715E-01  3.02890E-01
    20  1.03716E+00  5.34914E-03  0.005   0.51563    -5.18478E-01  3.03055E-01
    25  1.03699E+00  4.74850E-03  0.005   0.51577    -5.18548E-01  3.03007E-01
    30  1.03869E+00  5.50507E-03  0.005   0.51436    -5.17839E-01  3.03502E-01
    35  1.03980E+00  8.50778E-03  0.008   0.51343    -5.17376E-01  3.03825E-01
    40  1.04274E+00  8.05059E-03  0.008   0.51097    -5.16147E-01  3.04686E-01
    45  1.03252E+00  1.10480E-02  0.011   0.51952    -5.20426E-01  3.01699E-01
    50  1.03399E+00  1.01512E-02  0.010   0.51829    -5.19807E-01  3.02130E-01
```

It is interesting to note that the coefficient of variation fluctuates. This is most likely because the simulation once in a while produces a result, that is somewhat different from the others. This can happen because the simulation is based on the FORM result. If there is some probability content that is not covered well by the FORM approximation, the simulation will only hit this content a few times, and it will get a result that is different from the others each time it does so. This illustrates the weakness of basing a simulation upon an approximated result. The strength of doing this lies in the calculation speed when the FORM approximation is sound.

The number of lines in the table is controlled by use of the command: DEFINE RESULT-OPTION INTERMEDIATE-RESULTS.

It is possible to print and display the sample of correction values using the commands PRINT RESULT SAMPLE and DISPLAY RESULT DISTRIBUTION. These commands and the results are described in Section 3.6.1.

The simulation may be restarted from the previous result by using the command: RUN RESTART. The stop criteria may be changed before the run is restarted. This is useful e.g. for estimating the time a simulation need to run in order to produce a required accuracy on the result, or for continuing a simulation that did not produce the desired accuracy.

A conditional probability analysis is split into two analysis. The first for the intersection event and the second for the conditioning event. The sample cannot be printed in this case.

3.4 First Passage Probability and Results

3.4.1 Definition of a Stochastic Process for Calculation of First Passage Probability

A stochastic process is defined by assigning a random variable as the time derivative process of another random variable. Typically X and XDOT. The variables X and XDOT must be variables with type attribute Distribution, Fitted-Distribution or Generated distribution. The assignment of XDOT as the time derivative of X is done with the command:

```
ASSIGN CONTINUOUS-PROCESS TIME-DERIVATIVES X XDOT
```

The variable XDOT must have zero expectation. If no time variable is present in the model then the duration of the process is input by the command:

```
DEFINE CONTINUOUS-PROCESS DURATION 10800
```

The duration is 10800. If a time variable Time is present in the model, then the starting point and duration assigned to this variable is used:

```
CREATE VARIABLE Time 'Time variable' TIME
ASSIGN CONTINUOUS-PROCESS STARTING-TIME Time 1000
ASSIGN CONTINUOUS-PROCESS DURATION Time 10800
```

If assignment of starting time or duration is not explicitly done for Time, then the default values are used:

```
DEFINE CONTINUOUS-PROCESS STARTING-TIME 1000
DEFINE CONTINUOUS-PROCESS DURATION 10800
```

The above defined and assigned values may be undefined or unassigned in which case Proban issues an error message.

The first passage probability calculation is invoked by the command:

```
RUN CONTINUOUS-PROCESS-ANALYSIS FIRST-PASSAGE-PROBABILITY
```

Notice that Proban sets up the (nested) reliability analysis required to solve the problem. However, in order to manipulate differentiation options and convergence criteria for optimization algorithms, the user must know whether a nested reliability analysis is implied by the model at hand or not. The rule is rather simple. If the model includes a distribution variable not assigned a time derivative or assigned as a time derivative then Proban sets up a nested reliability analysis and the options for nested reliability analysis applies.

When a time variable is present in the model, the integration over time employs a trapezoidal rule. The integration is by default over the duration taken from the starting point. This interval may be reduced in order to capture the significant part of the time interval. The integration interval in the above example is restricted to the end of the interval by use of the command

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS INTEGRATION-INTERVAL 9000 10800
```

The number of points in the quadrature may be manipulated by the command

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS POINTS-IN-QUADRATURE 20
```

This puts 20 integration points in the interval 9000 to 10800.

If there is periodicity in the stochastic process, only one period needs to be integrated. The number of periods is input by the command

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS NUMBER-OF-TIME-SPLITS 2
```

The first passage probability may be the minimum of a number of independent realisations of the process. This number is entered by the command

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS MINIMUM-EXTREME-VALUE 3
```

It may be that the model requires a number of distribution variables other than the stochastic process variables to be integrated in the inner loop of the nested FORM analysis implied by the model. If the variable x is such a distribution variable then it is pushed to the inner loop by the command

```
ASSIGN SUB-LEVEL-INTEGRATION x ON
```

The outer loop integration of x is restored by the command

```
ASSIGN SUB-LEVEL-INTEGRATION x OFF
```

The calculation proceeds as for an ordinary probability calculation. The major difference is that importance factors are calculated only for the outer loop variables if a nested FORM analysis is implied by the model.

3.5 Crossing Rate and Results

3.5.1 Definition of a Stochastic Process for Calculation of Crossing Rate

A stochastic process is defined as above for the calculation of first passage probability.

If there is no time variable in the model, then neither starting time nor duration is made use of.

If there is a time variable in the model, then the crossing rate is calculated at the starting time for this variable, or, if not assigned, at the default starting time. If a duration is assigned to the time variable, then the crossing rate is averaged over the duration taken from the starting time. If a time variable is assigned a duration, then this value is used. If not, the default value is used. In order to avoid averaging both the duration assigned to Time and the default value must be turned off.

The following commands assures that the crossing rate is averaged over duration.

```
ASSIGN CONTINUOUS-PROCESS STARTING-TIME Time 1000
ASSIGN CONTINUOUS-PROCESS DURATION Time 10800
```

The following commands assures that the crossing rate is calculated at time 5000.

```
ASSIGN CONTINUOUS-PROCESS STARTING-TIME Time 5000
ASSIGN CONTINUOUS-PROCESS DURATION Time NONE
DEFINE CONTINUOUS-PROCESS DURATION NONE
```

The crossing rate calculation is invoked by the command:

```
RUN CONTINUOUS-PROCESS-ANALYSIS CROSSING-RATE
```

Notice that Proban sets up the FORM analysis required to solve the problem. If the model involves a distribution variable not assigned a time derivative or assigned as a time derivative then Proban sets up a nested FORM analysis and the options for nested FORM analysis applies.

When a time variable is present in the model and a duration is specified then time is integrated over by use of a trapezoidal rule. The integration is by default over the duration taken from the starting point. This interval may be reduced in order to capture the significant part of the time interval:

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS INTEGRATION-INTERVAL 9000 10800
```

The number of points in the quadrature may be manipulated by the command

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS POINTS-IN-QUADRATURE 20
```

This puts 20 integration points in the interval 9000 to 10800.

If there is periodicity in the stochastic process, only one period needs to be integrated. The number of periods is input by the command

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS NUMBER-OF-TIME-SPLITS 2
```

The calculation proceeds as for an ordinary probability calculation. The major difference is that importance factors are calculated only for the outer loop variables if a nested FORM analysis is implied by the model.

3.6 Distribution Analysis and Results

Performing a distribution analysis requires the following steps, after the model has been specified:

1 Select the method to be used for distribution analysis, using SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS. The default method is MONTE-CARLO-SIMULATION when Proban starts from a new database.

2 Define the desired options for the chosen method and/or general analysis options. These options are explained in the DEFINE command. The default options will be sufficient in most cases.

3 ASSIGN SENSITIVITY-CALCULATION to the required parameters and/or decide the extent of sensitivity calculation using DEFINE ANALYSIS-OPTION SENSITIVITY.

4 Run the analysis using RUN DISTRIBUTION-ANALYSIS.

5 Present the results using PRINT RESULT, DISPLAY RESULT and PLOT.

The different analysis methods are described in separate sections, using the examples from Section 3.1

Monte Carlo simulation and latin hypercube simulation are treated together because the presentation of results is the same for both.

3.6.1 Monte Carlo and Latin Hypercube Simulation

The default distribution sampling method is Monte carlo simulation. This is a straightforward sampling method, that repeatedly samples all the random variables in the model and calculates the target value from them.

Latin hypercube simulation follows the same principle, but uses a stratified sampling technique, that is usually more economical.

These methods may be used to calculate parametric sensitivity values, but not to calculate importance factors. Each sensitivity calculation requires numerical differentiation, and consequently the sampling of an extra value for each differentiation done. This can increase computation time considerably. In the run listed below, 5 sensitivity values are simulated. This increases the time the simulation need to run approximately a factor of 6.

The number of simulations can be controlled using the command: DEFINE DISTRIBUTION-SIMULATION. The cpu time usage cannot be controlled.

Consider the calculation of the distribution of the Net Present Value in Example 3.2. The following command produces a simulation including a sensitivity calculation with respect to the mean of all variables (the messages generated by Proban are included).

```
ASSIGN SENSITIVITY-CALCULATION VARIABLE *-Mean*
      Assigned sensitivity calculation to the Mean of E1
      Assigned sensitivity calculation to the Mean of E2
```

```
      Assigned sensitivity calculation to the Mean of I1
      Assigned sensitivity calculation to the Mean of I2
      Assigned sensitivity calculation to the Mean of S
RUN DISTRIBUTION-ANALYSIS NPV

      Starting Distribution Analysis of: NPV

      Starting Monte Carlo simulation

      Stopping after 1000 simulations
      Simulating 5 sensitivity values
           250 simulations completed.
           500 simulations completed.
           750 simulations completed.
          1000 simulations completed.

      Number of simulations:                1000
      Estimated Mean:                 4.96924E+03
      Estimated Standard Deviation:   9.23969E+03
      Estimated Skewness:                  0.121
      Estimated Kurtosis:                  2.773
      Normal distribution fit to simulation of: NPV
      stored in a variable called: Normal-Fit
      Hermit-trans distribution fit to simulation of: NPV
      stored in a variable called: Hermite-Fit

SAVE RESULT MCS-NPV 'Monte Carlo simulation of the Net Present Value'
      MCS-NPV is now the selected result
```

After the analysis, the result was saved under the name MCS-NPV.

The fitted distributions are based on the estimated moments. From the skewness and kurtosis it can be seen that the distribution fits well to a normal distribution (which has skewness 0 and kurtosis 3).

This may also be checked using the DISPLAY command:

```
SET GRAPH HISTOGRAM FILLING HOLLOW
DISPLAY RESULT DISTRIBUTION ( INCLUDE Empirical Normal-Fit Hermite-Fit )
    LOOP
    DENSITY
    # SET DRAWING GRID ON
    # PLOT
    DISTRIBUTION
    END
PLOT
```

These commands generate the following two plots:

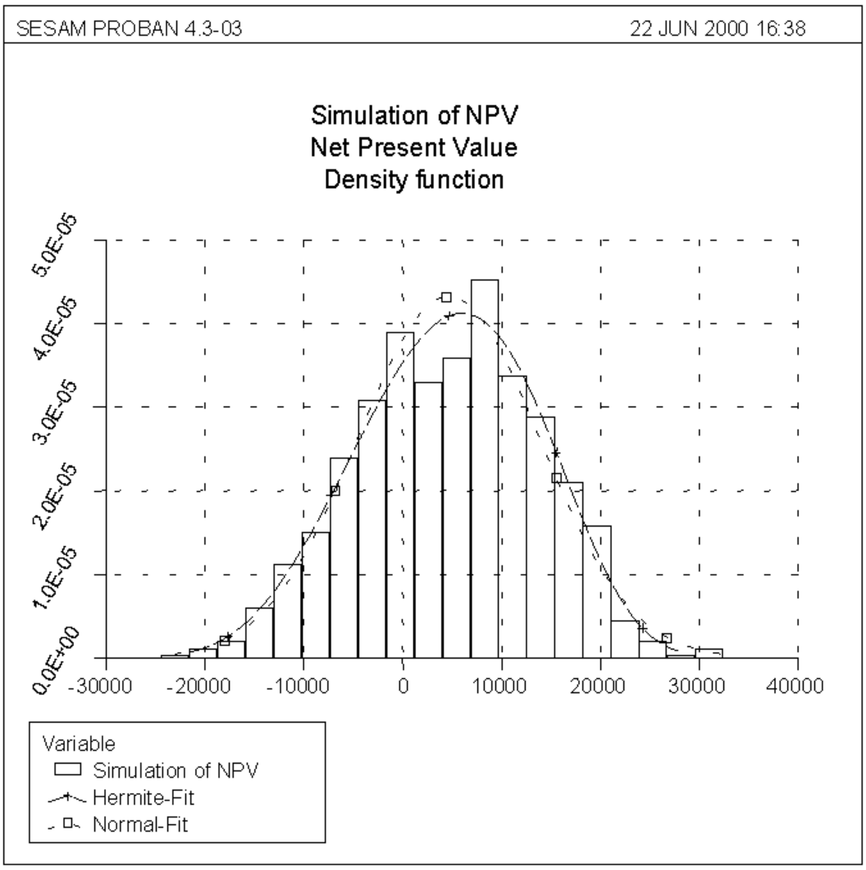


Figure 3.5 Histogram of NPV with fitted distributions

As can be seen, the difference between the normal distribution fit and hermite transformation distribution fit is small.

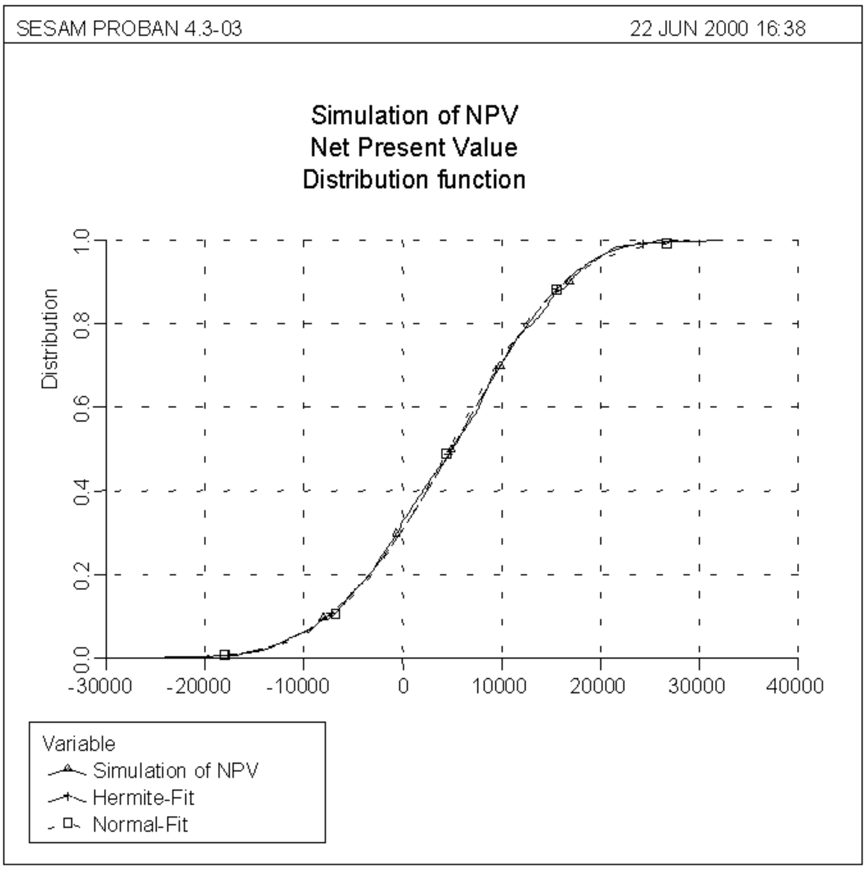


Figure 3.6 Empirical distribution function for NPV with fitted distributions

The result print includes the options: ANALYSIS-SETTINGS, SUMMARY, ALL, SENSITIVITY and SAMPLE.

The print of ANALYSIS-SETTINGS contains the analysis settings used with the analysis plus the cpu time usage and the time and date of the run.

The summary print contains simply the estimated moments:

```
PRINT RESULT SUMMARY
```

```
+------------------------------------------------------------------+
! Result name    : MCS-NPV                                           !
!                  Monte carlo simulation of the Net Present Value !
! Distribution of: NPV                                               !
!                  Net Present Value                                 !
! Analysis method: Monte Carlo simulation                            !
+------------------------------------------------------------------+
```

```
                  Final results after 1000 simulations

          Mean        :  4.96924E+03    Skewness:  1.20695E-01
          Standard-Dev:  9.23969E+03    Kurtosis:  2.77288E+00
```

The print of sensitivity results contains four tables, similar to the table presented for directional simulation. There is one table with sensitivity values for each of the four moments:

```
PRINT RESULT SENSITIVITY

             Parametric sensitivity result for Mean = 4969.2439404

   Variable     Type         Parameter      Value    dMean/dPar  Measure
   ---------------------------------------------------------------------
   I1           Triangle     Mean          7.500E+04  9.292E-01  6.97E+03
   E1           Lognormal    Mean          5.000E+03 -9.082E-01 -4.54E+02
   I2           Triangle     Mean          5.000E+04  8.174E-01  4.09E+03
   E2           Lognormal    Mean          1.000E+04 -8.272E-01 -8.27E+02
   S            Normal       Mean          1.000E+04  8.264E-01  8.26E+02


      Parametric sensitivity result for Standard Deviation = 9239.6947365

   Variable     Type         Parameter      Value    dStDv/dPar  Measure
   ---------------------------------------------------------------------
   I1           Triangle     Mean          7.500E+04 -1.142E-02 -8.56E+01
   E1           Lognormal    Mean          5.000E+03  2.365E-04  1.18E-01
   I2           Triangle     Mean          5.000E+04 -1.401E-02 -7.00E+01
   E2           Lognormal    Mean          1.000E+04  1.372E-03  1.37E+00
   S            Normal       Mean          1.000E+04 -1.834E-15 -1.83E-12


          Parametric sensitivity result for Skewness = 0.12069534213

   Variable     Type         Parameter      Value    dSkew/dPar  Measure
   ---------------------------------------------------------------------
   I1           Triangle     Mean          7.500E+04 -3.889E-05  -0.29169
   E1           Lognormal    Mean          5.000E+03  1.146E-06   0.00057
   I2           Triangle     Mean          5.000E+04 -5.882E-05  -0.29412
   E2           Lognormal    Mean          1.000E+04  9.519E-07   0.00095
   S            Normal       Mean          1.000E+04 -4.203E-19   0.00000


           Parametric sensitivity result for Kurtosis = 2.7728764069

   Variable     Type         Parameter      Value    dKurt/dPar  Measure
   ---------------------------------------------------------------------
   I1           Triangle     Mean          7.500E+04 -1.298E-05  -0.09733
   E1           Lognormal    Mean          5.000E+03 -6.829E-07  -0.00034
   I2           Triangle     Mean          5.000E+04 -2.853E-06  -0.01427
   E2           Lognormal    Mean          1.000E+04  2.508E-07   0.00025
```

```
   S             Normal       Mean          1.000E+04  1.714E-18   0.00000
```

A change in the mean will cause a shift in the distribution (affecting the profit), while a change in the standard deviation will cause a change in the slope of the distribution function at the centre of the distribution (affecting the risk). See also Figure 2.9 and Figure 2.10.

The sensitivity measure is useful for getting an overview of the effect of the different parameters. The measure is described in Section 2.8.

The ALL option gives the print shown above plus a list of intermediate simulation results and a table showing the empirical distribution.

The table of intermediate results is useful for checking if the simulation has stabilised. If the mean or standard deviation fluctuate it may be necessary to continue the simulation (see RUN RESTART). The skewness and kurtosis can be expected to fluctuate. They describe the tail behaviour of the distribution, and require a very large number of simulations in order to be estimated accurately.

```
                  Intermediate simulation results:

      NoSim     Mean     Standard-Dev Skewness  Kurtosis
     ---------------------------------------------------
        100  5.38336E+03 9.867633E+03    0.4084    3.1206
        200  4.94207E+03 9.639992E+03    0.1910    2.9822
        300  4.97684E+03 9.678793E+03    0.1232    2.9344
        400  4.99982E+03 9.648684E+03    0.1564    2.8548
        500  4.99812E+03 9.561949E+03    0.1082    2.7310
        600  5.17163E+03 9.439078E+03    0.1123    2.7143
        700  4.95297E+03 9.474204E+03    0.0946    2.7041
        800  5.00293E+03 9.413223E+03    0.0857    2.7489
        900  5.08599E+03 9.295718E+03    0.0955    2.7624
       1000  4.96924E+03 9.239695E+03    0.1207    2.7729
```

The number of lines in the table is controlled by use of the command DEFINE PRESENTATION RESULT INTERMEDIATE-SIMULATIONS.

The table of the empirical distribution function has the following contents.

```
      Empirical Distribution

         Fractile      Prob
     -----------------------
     -2.188292616E+04  0.001
     -1.593654735E+04  0.010
     -1.012259806E+04  0.050
     -6.582621465E+03  0.100
     -4.363203073E+03  0.150
     -2.738950028E+03  0.200
     -1.270994459E+03  0.250
     -2.631476443E+02  0.300
      1.978407038E+03  0.400
```

```
4.159150481E+03  0.500
7.121619541E+03  0.600
9.876500914E+03  0.700
1.131511362E+04  0.750
1.319665845E+04  0.800
1.496162676E+04  0.850
1.736876892E+04  0.900
2.045195947E+04  0.950
2.630603369E+04  0.990
3.618389864E+04  0.999
```

The final print option gives a print of the whole sample, the first column showing the values in the order they were sampled, and the second column showing the values in increasing order. The length of this print is usually very large (a typical sample is 1000 values, producing more than 1000 lines of print), so be careful with this one.

The following lists the beginning and end of a print of a sample

```
PRINT RESULT SAMPLE
```

```
+-------------------------------------------------------------------+
! Result name    : MCS-NPV                                          !
!                  Monte carlo simulation of the Net Present Value !
! Distribution of: NPV                                              !
!                  Net Present Value                                !
! Analysis method: Monte Carlo simulation                           !
+-------------------------------------------------------------------+


                     Simulated observations

          SimNo   Observation         Sorted
         ----------------------------------------
              1  1.091962657E+03 -2.188313507E+04
              2  4.877797384E+03 -2.167421873E+04
              3 -2.817265088E+02 -1.898907363E+04
              4  6.398132552E+03 -1.813028770E+04
              5 -8.561568457E+03 -1.749456502E+04
              .  .                 .
              .  .                 .
              .  .                 .
            997  5.078346459E+03  2.883378727E+04
            998 -1.332132339E+04  3.004064064E+04
            999  8.859116186E+03  3.060224454E+04
           1000 -2.935722632E+03  3.618948588E+04
```

The simulation may be restarted using the RUN RESTART command, continuing from the previous result. The number of simulations to be done can be changed before the restart.

Consider now the situation in Example 3.2 after 6 months, where the manager has obtained information that the income after the first year will exceed 70000. The updated distribution for the Net Present value is calculated as follows (showing also the Proban messages):

```
RENAME VARIABLE Hermite-Fit Original-Fit
      Renamed variable Hermite-Fit to Original-Fit
RUN DISTRIBUTION-ANALYSIS CONDITIONED NPV SINGLE-EVENT I1 > 70000

      Starting Distribution Analysis of: NPV given I1 > 7000

      Starting Monte Carlo simulation

      Stopping after 1000 simulations
      Simulating 5 sensitivity values
           250 simulations completed.
           500 simulations completed.
           750 simulations completed.
          1000 simulations completed.


      Number of simulations:                1000
      Number in conditioning event:          786
      Estimated Mean:                7.34214E+03
      Estimated Standard Deviation:  8.19665E+03
      Estimated Skewness:                 -0.013

      Estimated Kurtosis:                  2.663
      Normal distribution fit to simulation of: NPV
      stored in a variable called: Normal-Fit
      Hermit-trans distribution fit to simulation of: NPV
      stored in a variable called: Hermite-Fit


RENAME VARIABLE Hermite-Fit Updated-Fit
      Renamed variable Hermite-Fit to Updated-Fit
SAVE RESULT Updated 'Distribution of NPV given I1 > 70000'
      Updated is now the selected result
```

Proban prints the number of hits in the conditioning event, as well as the usual information. The distribution cannot be estimated if there is no hit in the conditioning event.

Note that the fitted distributions were renamed, so that they will not be overwritten during the next analysis. This enables a comparison between the original distribution and the updated distribution:

```
SET TITLE 'Updating of NPV distribution' 'Income after first year > 70000' ;
DISPLAY RESULT DISTRIBUTION ( ONLY Original-Fit Updated-Fit ) DISTRIBUTION
```

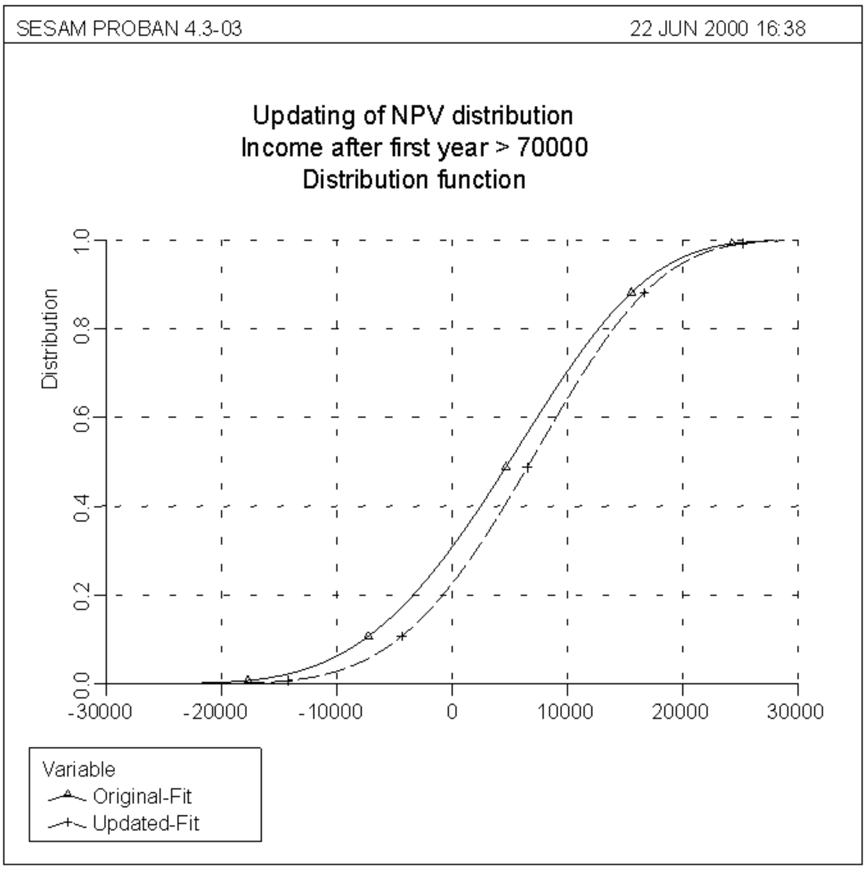


Figure 3.7 Comparison between original and updated distribution of NPV

The income still has the same upper limit, so the upper limit of the distribution has not been changed. Instead, the centre and lower tail is shifted, so that the probability of a loss now is about 0.2.

3.6.2 Mean Value Based FORM

The mean value based FORM method gives an estimate of the distribution function of a variable. The options for the method are controlled by the command: DEFINE MEAN-VALUE-FORM (see the explanation of this command). The default options will usually suffice.

Sensitivity calculation and conditional distribution analysis cannot be done using the mean value based FORM method.

Running this method on the Net Present Value of Example 3.2 using the default options gives the following messages from Proban.

```
SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS MEAN-VALUE-FORM
RUN DISTRIBUTION-ANALYSIS NPV
```

```
Starting Distribution Analysis of: NPV

Starting Mean Value based FORM calculation

Using 19 points from probability 1.0E-02 to 0.99
Mean Value based FORM calculation completed
```

The result may be printed:

```
PRINT RESULT ALL
```

```
+-------------------------------------+
! Distribution of: NPV                !
!                  Net Present Value !
! Analysis method: Mean Value FORM   !
+-------------------------------------+


     Fractile        Prob       Beta
 ----------------------------------
 -1.645756331E+04  0.010000   2.3263
 -1.441564890E+04  0.019326   2.0679
 -1.220139988E+04  0.035196   1.8094
 -9.839270797E+03  0.060463   1.5509
 -7.362152817E+03  0.098107   1.2924
 -4.810213974E+03  0.150584   1.0339
 -2.229028106E+03  0.219037   0.7754
  3.328901273E+02  0.302590   0.5170
  2.826943311E+03  0.398017   0.2585
  5.207437625E+03  0.500000   0.0000
  7.587310473E+03  0.601983  -0.2585
  1.007949922E+04  0.697410  -0.5170


  1.263830995E+04  0.780963  -0.7754
  1.521514507E+04  0.849416  -1.0339
  1.776148968E+04  0.901893  -1.2924
  2.023176963E+04  0.939537  -1.5509
  2.258581651E+04  0.964804  -1.8094
  2.479073870E+04  0.980674  -2.0679
  2.682208115E+04  0.990000  -2.3263
```

The only other available print option is ANALYSIS-SETTINGS.

The result may also be displayed together with other distributions (in this case the hermite fit to the simulated distribution of NPV from the previous section):

```
SET DRAWING FONT-SIZE RELATIVE 1.5
DISPLAY RESULT DISTRIBUTION ( ONLY Mean-V-FORM Original-Fit ) DISTRIBUTION
```

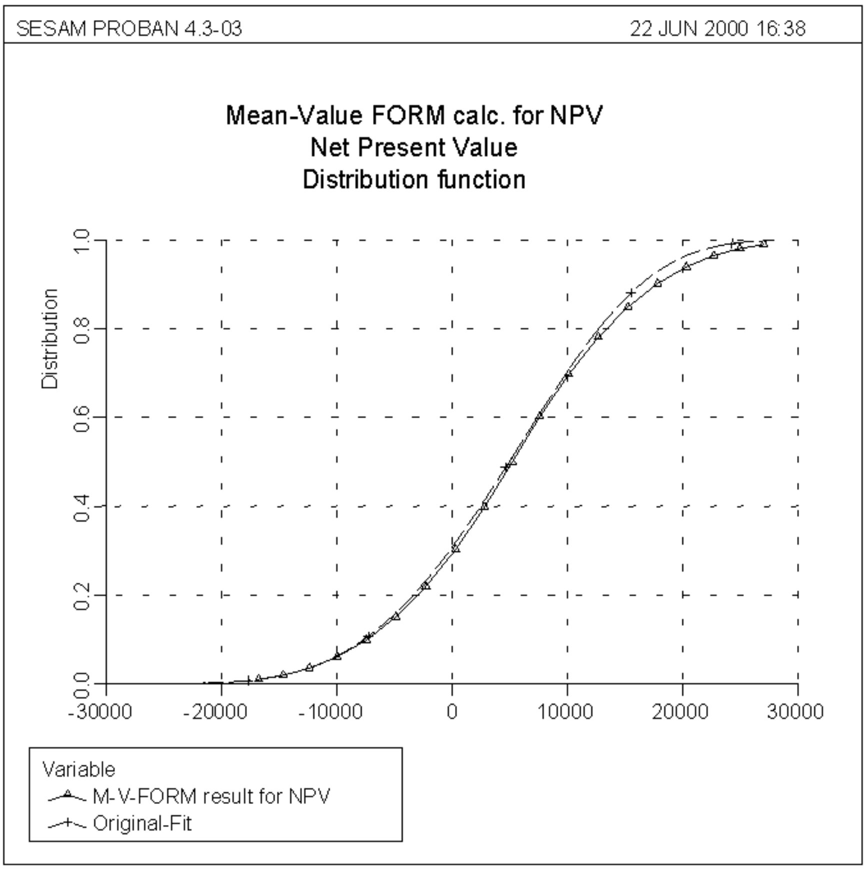


Figure 3.8 Mean value based FORM distribution for NPV with Hermite fit

In this case the mean value based FORM result is quite accurate. This is not necessarily the case.

3.7 Deterministic Analysis and Results

It is often helpful to calculate the value of a variable or an event function at a specified point in order to verify the formulation of a stochastic model. In Proban this is achieved through the RUN DETERMINISTIC-ANALYSIS command.

The following command calculates the value of the variable x at the mean of the random variables in the model:

```
RUN DETERMINISTIC-ANALYSIS VARIABLE X MEAN-VALUE
```

The following command calculates the value of the variable at a point modified from the mean:

```
RUN DETERMINISTIC-ANALYSIS VARIABLE X MODIFIED MEAN-BASED ...
```

and then entering the modifications. The analysis of a variable can also be median based.

The following command calculates the event function Beam-Fail at the starting point for a FORM/SORM analysis:

```
RUN DETERMINISTIC-ANALYSIS EVENT Beam-Fail STARTING-POINT
```

The following command calculates an event function at the U-space origin:

```
RUN DETERMINISTIC-ANALYSIS EVENT Beam-Fail USPACE-ORIGIN
```

and produces the result:

```
Value of event Beam-Fail: 30136.798306 (False)
```

The points of the union event Beam-Fail have negative function values. Therefore, if the value was negative, then the assertion that the point is in the Beam-Fail event it would be (True). However, the point takes a positive value, and therefore the assertion is (False).

3.8 Parameter Study Analysis and Results

It is often desirable to monitor the development of a target value as a function of a parameter in the model (e.g. as a function of time). This can be done in Proban by use of the parameter study facility.

1 The steps in performing a parameter study are:

2 Enter the model into Proban.

3 Assign a parameter study to parameter, specifying the desired values. This is done using the ASSIGN PARAMETER-STUDY command.

4 If necessary, make sure that a parameter study will be run by entering the command: DEFINE ANALYSIS-OPTION PARAMETER-STUDY ON. The default status is ON, so this is only necessary if the current status has been set to OFF.

5 Run the analysis, using the RUN command. One analysis will be performed for each parameter value.

6 Present the results as a function of the parameter using PRINT RESULT PARAMETER-STUDY and DISPLAY RESULT PARAMETER-STUDY, and/or present the individual analysis results using PRINT RESULT and DISPLAY RESULT.

If the next analysis is to be done without using the parameter study, there are two options available: DEFINE ANALYSIS-OPTION PARAMETER-STUDY OFF (temporarily disabling the parameter study) and ASSIGN PARAMETER-STUDY <current-parameter> EXCLUDE * (removing the assignment).

When a parameter study has been run, the main results and importance factors (if available) can be printed and displayed as a function of the parameter. The available main results are listed in Section 2.10.

As an example, consider Example 3.2, described in Section 3.1.

The manager wishes to investigate the connection between the Net Present Value of the project and the required rate of return. Two parameter studies are done. One for the distribution of the NPV, and one for the probability of a loss.

The parameter study for r in the range from 1% to 15% is assigned as follows:

```
DEFINE PARAMETER-STUDY r GROUP 0.01 0.15 0.01
      Defined parameter study for r using the values 1.0E-02 2.0E-02 3.0E-02
      4.0E-02 5.0E-02 6.0E-02 7.0E-02 8.0E-02 9.0E-02 0.1 0.11 0.12 0.13 0.14
      0.15
```

For this analysis, no parametric sensitivity values are required:

```
ASSIGN SENSITIVITY-CALCULATION VARIABLE EXCLUDE *
      Deassigned sensitivity calculation for the Mean of E1
      Deassigned sensitivity calculation for the Mean of E2
      Deassigned sensitivity calculation for the Mean of I1
      Deassigned sensitivity calculation for the Mean of I2
      Deassigned sensitivity calculation for the Mean of S
```

The distribution analysis will take some time. Proban gives the standard analysis message for each analysis, and shows the parameter value used in the analysis. Only parts of these messages are shown here.

```
SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS MONTE-CARLO-SIMULATION
RUN DISTRIBUTION-ANALYSIS NPV


      Starting Distribution Analysis of: NPV


      Parameter study: r =  0.100000E-01


      Starting Monte Carlo simulation

      Stopping after 1000 simulations
           250 simulations completed.
           500 simulations completed.
           750 simulations completed.
          1000 simulations completed.

      Number of simulations:              1000
      Estimated Mean:                1.85046E+04
      Estimated Standard Deviation:  1.07176E+04
      Estimated Skewness:                 -0.018
      Estimated Kurtosis:                  2.812


      Parameter study: r =  0.200000E-01


      Starting Monte Carlo simulation
```

and so on ...

When using a simulation in a parameter study, Proban will use the same seeds in each analysis. This will make the results correlated, but will remove the random fluctuation between analyses. If this was not done, comparison between results for different parameter values would be very difficult.

A simulated parameter study may be continued using the command: RUN RESTART.

The main results can be printed and displayed, as shown below:

```
PRINT RESULT PARAMETER-STUDY MAIN-RESULT ( Mean Stand-Dev Skewness Kurtosis )
```

```
                  +------------------------------------------+
                  ! Distribution of: NPV                     !
                  !                  Net Present Value       !
                  ! Analysis method: Monte Carlo simulation  !
                  +------------------------------------------+


                             Parameter study  :  r


      Value        : r
      Mean         : Estimated Mean
      Standard-Dev : Estimated Standard Deviation
      Skewness     : Estimated Skewness
      Kurtosis     : Estimated Kurtosis


         Value         Mean       Standard-Dev   Skewness     Kurtosis
      ----------------------------------------------------------------
       1.00000E-02  1.79527E+04  1.06288E+04    -0.092        2.604
       2.00000E-02  1.63237E+04  1.04551E+04    -0.093        2.604
       3.00000E-02  1.47354E+04  1.02863E+04    -0.093        2.605
       4.00000E-02  1.31864E+04  1.01222E+04    -0.093        2.605
       5.00000E-02  1.16753E+04  9.96274E+03    -0.094        2.605
       6.00000E-02  1.02008E+04  9.80762E+03    -0.094        2.606

       7.00000E-02  8.76168E+03  9.65672E+03    -0.094        2.606
       8.00000E-02  7.35671E+03  9.50988E+03    -0.095        2.606
       9.00000E-02  5.98475E+03  9.36697E+03    -0.095        2.606
       1.00000E-01  4.64472E+03  9.22783E+03    -0.095        2.606
       1.10000E-01  3.33555E+03  9.09233E+03    -0.095        2.607
       1.20000E-01  2.05624E+03  8.96034E+03    -0.095        2.607
       1.30000E-01  8.05832E+02  8.83174E+03    -0.096        2.607
       1.40000E-01 -4.16599E+02  8.70640E+03    -0.096        2.607
       1.50000E-01 -1.61194E+03  8.58423E+03    -0.096        2.607
```

The only parameter that varies considerably is the mean. The standard deviation decreases slightly when r increases

The mean with confidence limits is displayed as a function of r (and a file copy is created):

```
SET DRAWING FONT-SIZE RELATIVE 1.5
```

```
SET GRAPH LINE-OPTIONS MARKER OFF
DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT *Mean*
PLOT
SET GRAPH LINE-OPTIONS MARKER ON
```

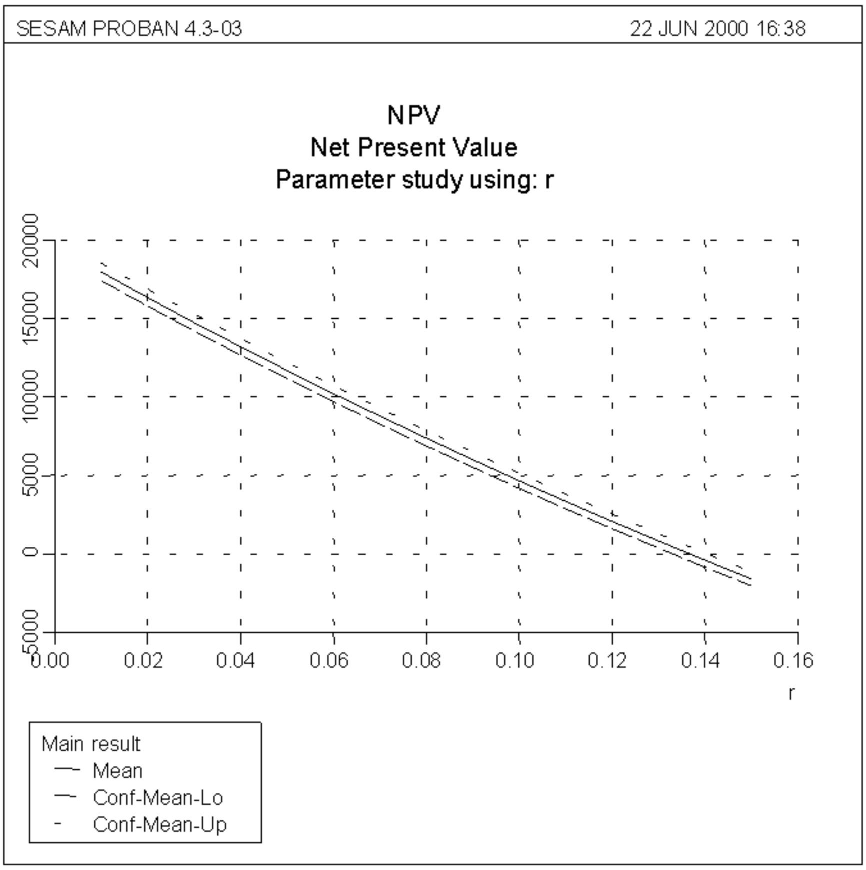


Figure 3.9 Parameter study of mean of NPV with respect to internal rate of return

To get a visual impression of the development of the whole distribution, here is a display of four of the fitted distributions:

```
CREATE VARIABLE
    LOOP
    NPV_01 'Fit to NPV with r = 0.01' DISTR Normal Mean-StD  17952.7  10628.8
    NPV_05 'Fit to NPV with r = 0.05' DISTR Normal Mean-StD  11675.3   9962.74
    NPV_10 'Fit to NPV with r = 0.10' DISTR Normal Mean-StD   4644.72  9227.83
    NPV_15 'Fit to NPV with r = 0.15' DISTR Normal Mean-StD  -1611.94  8584.23E
    END
DISPLAY DISTRIBUTION ONLY NPV_&&
    LOOP
    DENSITY
```

```
DISTRIBUTION
END
```

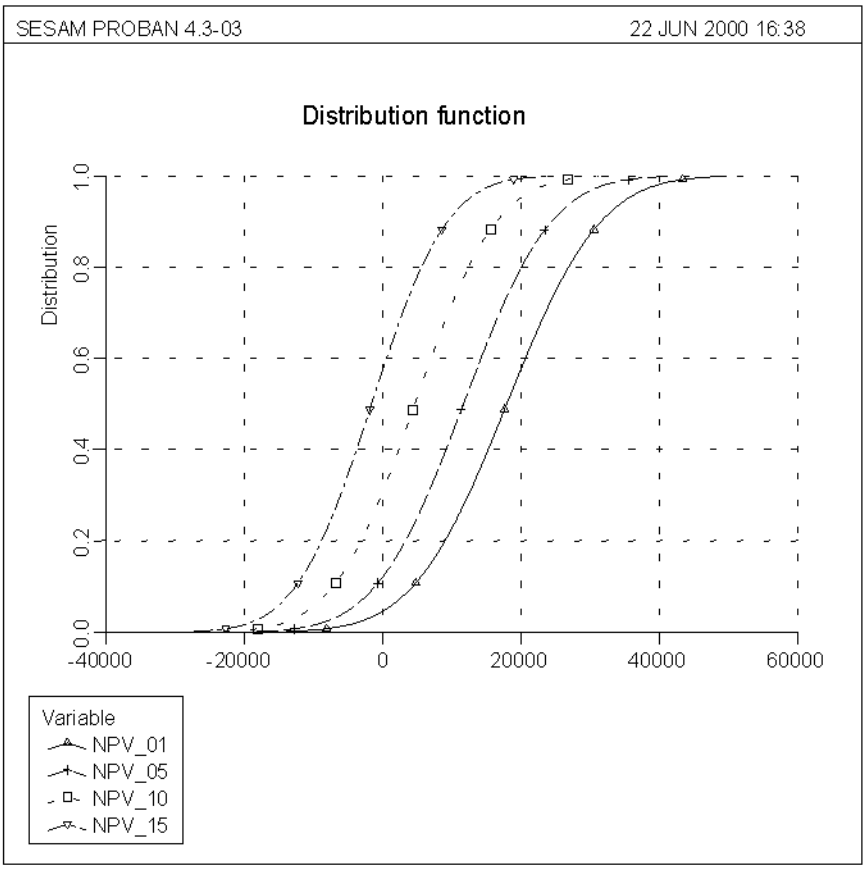


Figure 3.10 The distribution of NPV for different rates of return

The density function plot clearly shows the change in both mean and standard deviation:

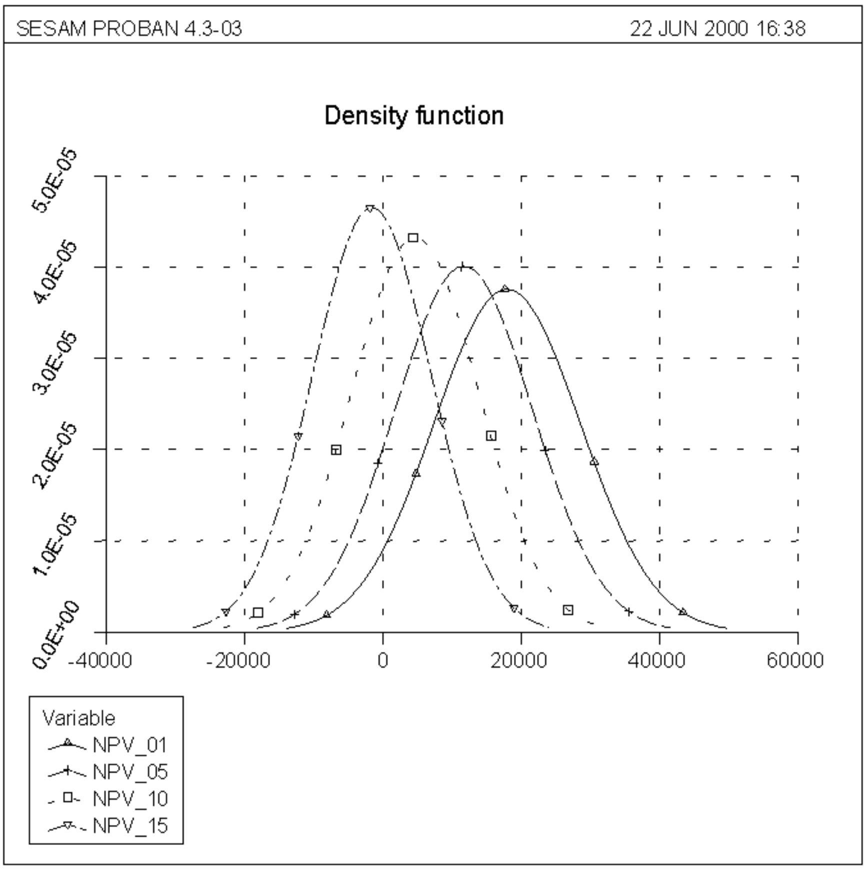


Figure 3.11 The density of NPV for different rates of return

The 15 individual results may also be examined independently, by selecting one (in DISPLAY) or any number (in PRINT) of the parameter values with the usual DISPLAY RESULT and PRINT RESULT command.

The probability of a loss is examined using a FORM analysis:

```
SELECT ANALYSIS-METHOD PROBABILITY FORM
RUN PROBABILITY-ANALYSIS SINGLE-EVENT NPV < 0

      Starting Probability Analysis of: NPV < 0.0


      Parameter study: r =  0.100000E-01


      Starting FORM calculation

      Starting linearization of:
```

```
Single event:  NPV < 0.0
Linearization completed
Calculating importance factors

FORM  Reliability index:       1.6589
FORM  Probability:        4.85693E-02


Parameter study: r =  0.200000E-01


Starting FORM calculation
```

and so on ...

The results can be presented in a table:

```
PRINT RESULT PARAMETER-STUDY MAIN-RESULT *
```

```
   Value      Beta-FORM    Prob-FORM   Log10P-FORM
---------------------------------------------------
 1.00000E-02    1.6589     4.85693E-02   -1.3136
 2.00000E-02    1.5333     6.26036E-02   -1.2034
 3.00000E-02    1.4092     7.93830E-02   -1.1003
 4.00000E-02    1.2863     9.91689E-02   -1.0036
 5.00000E-02    1.1640     1.22202E-01   -0.9129
 6.00000E-02    1.0420     1.48698E-01   -0.8277
 7.00000E-02    0.9198     1.78836E-01   -0.7475
 8.00000E-02    0.7969     2.12752E-01   -0.6721
 9.00000E-02    0.6728     2.50535E-01   -0.6011
 1.00000E-01    0.5470     2.92197E-01   -0.5343
 1.10000E-01    0.4188     3.37676E-01   -0.4715
 1.20000E-01    0.2876     3.86808E-01   -0.4125
 1.30000E-01    0.1527     4.39316E-01   -0.3572
 1.40000E-01    0.0131     4.94779E-01   -0.3056
 1.50000E-01   -0.1285     5.51104E-01   -0.2588
```

and they can also be displayed, as with the main results above.

```
SET DRAWING FONT-SIZE RELATIVE 1.5
DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT ONLY Prob*
```

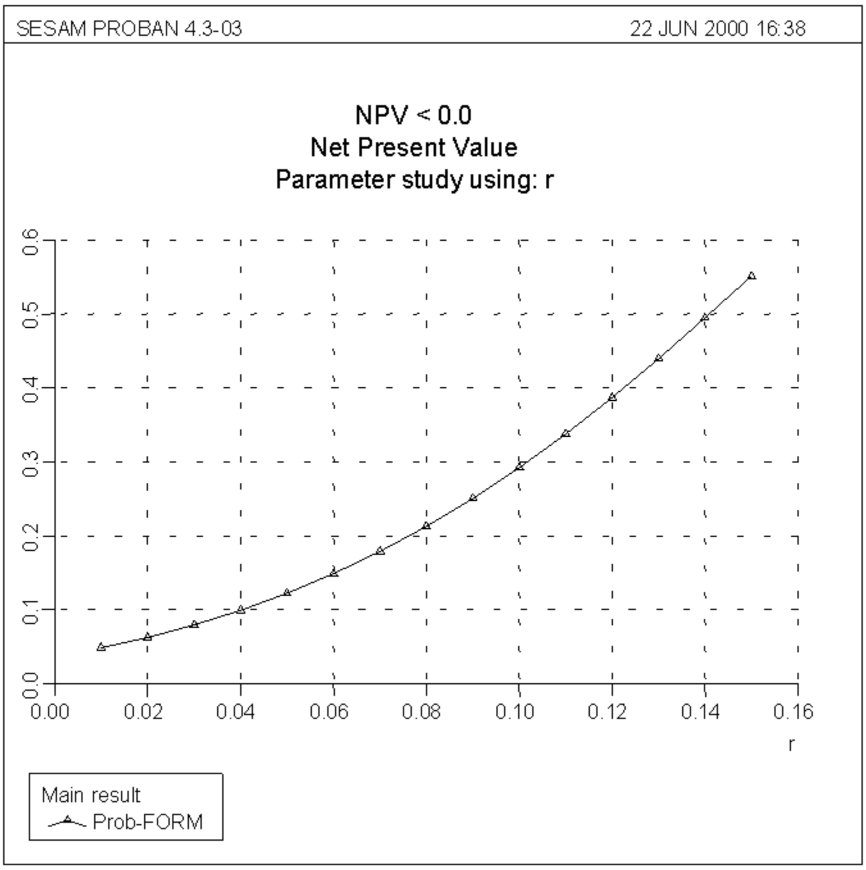


Figure 3.12 The probability of a loss different rates of return

The importance factor pie charts can be displayed simultaneously for a selection of parameter values:

It is also possible to print and display the variation of the importance factors with the internal rate of return:

```
                +------------------------------------+
                ! Probability of : NPV < 0.0         !
                !                  Net Present Value !
                ! Analysis method: SORM              !
                +------------------------------------+


                         Parameter study  :  r

      Value        : r
      I1           : Importance factor for I1
      I2           : Importance factor for I2
      S            : Importance factor for S
```

```
ImpGroup-1   : Importance factor for Group number 1

  Value          I1           I2           S        ImpGroup-1
----------------------------------------------------------------
 1.00000E-02    35.4         53.8         3.3          7.5
 2.00000E-02    35.5         54.3         3.2          7.1
 3.00000E-02    35.5         54.7         3.0          6.7
 4.00000E-02    35.6         55.1         2.9          6.4
 5.00000E-02    35.6         55.4         2.8          6.2
 6.00000E-02    35.7         55.5         2.8          6.0
 7.00000E-02    35.8         55.5         2.8          5.9
 8.00000E-02    36.0         55.4         2.7          5.8
 9.00000E-02    36.2         55.2         2.7          5.8
 1.00000E-01    36.5         54.9         2.8          5.9
 1.10000E-01    36.8         54.4         2.8          6.0
 1.20000E-01    37.1         53.8         2.9          6.2
 1.30000E-01    37.5         53.0         3.1          6.4
 1.40000E-01    37.9         52.1         3.2          6.7
 1.50000E-01    38.5         52.2         3.1          6.3
```

In this case there is practically no difference in the importance factors for different values of the rate of return.

If the importance factors are mapped across the distribution of the NPV, they will often be seen to change considerably. This can be done by finding the probability of the event: NPV < x as a function of x.

The same type of parameter study may be used to map the distribution function, using a probability analysis method.

The following commands will do the trick:

```
CREATE VARIABLE x ' ' FIXED 0
CREATE VARIABLE NPVx ' ' FUNCTION Difference NPV x
CREATE EVENT NPVx ' ' SINGLE NPVx < 0  % NPV - x < 0 is identical to: NPV < x
ASSIGN PARAMETER-STUDY x ONLY GROUP -25000 30000 1000
SELECT ANALYSIS-METHOD PROBABILITY SORM PARABOLIC
RUN PROBABILITY-ANALYSIS NPVx
SET DRAWING FONT-SIZE RELATIVE 1.5
DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR *
```

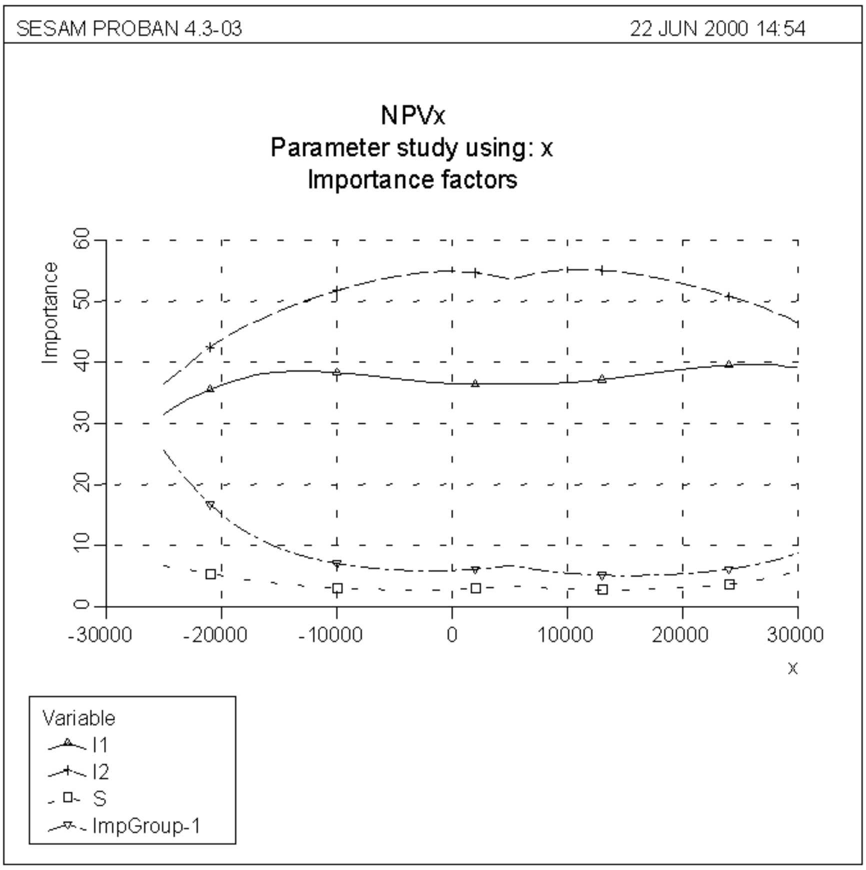


Figure 3.13 The importance factors across the distribution of NPV

The spike in the middle is caused by the application of FORM/SORM to a model containing the non-differentiable density function of the triangle distribution.

The spikes will disappear if the triangle distributions are changed to Beta distributions with the same mean, standard deviation and limits, or if Directional simulation is used.

One can also display importance factor pie charts simultaneously for a selection of parameter values from a parameter study by using the command:

```
SET DRAWING FONT-SIZE RELATIVE 1.0
DISPLAY RESULT MAIN-RESULT ONLY ( -15000 0 15000 )
```

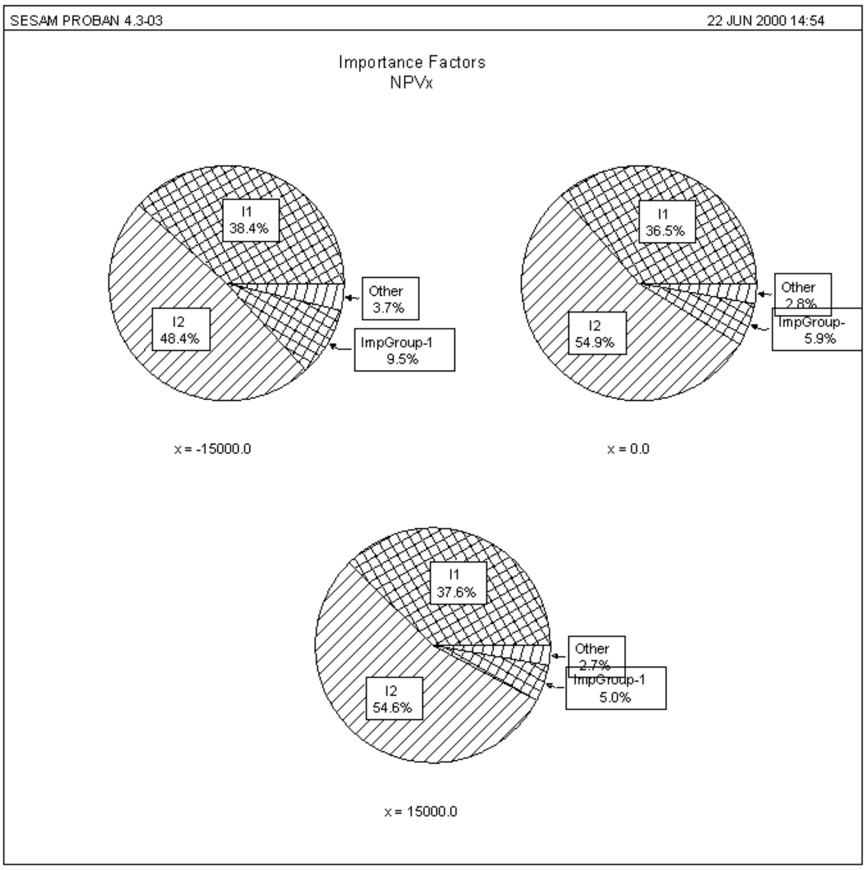


Figure 3.14 Multiple pie charts for parameter study of importance factors

While the results are available, it is instructing to see the distribution, as calculated with FORM and SORM:

```
DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT ONLY Prob*
```

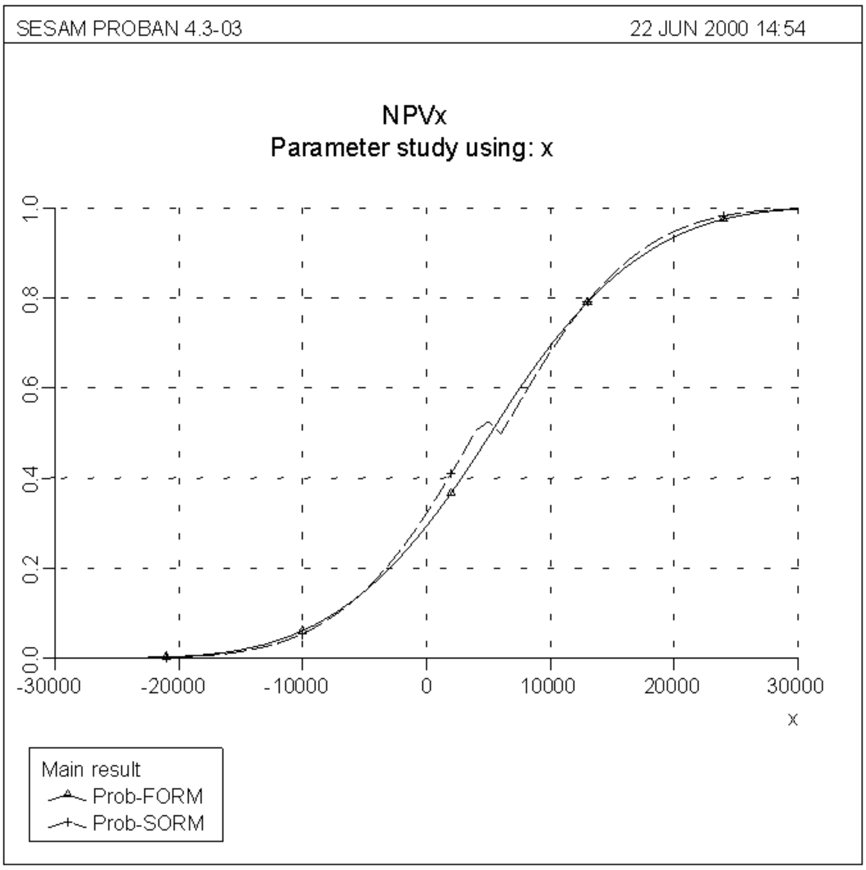


Figure 3.15 The distribution of NPV calculated by FORM and SORM

The SORM distribution has a nasty drop at the middle. This is not a Proban error, it is caused by an improper usage of SORM. Again, the triangle distributions used here have a non-differentiable density at the middle of the distributions. If similar Beta distributions were used, the SORM result would give a correct distribution function (try it!).

3.9 Distributions

Proban contains an extensive list of distributions, that can be used to model uncertainty. The list includes 21 continuous distributions, two discrete distributions and a spline distribution, that fits a distribution function to a set of input points. Also, a number of the continuous distributions can be used to fit data (generated by Proban) by use of maximum likelihood fits or least square fits.

Section 3.9.1 describes the distributions, that are available in Proban.

Section 3.9.2 gives an example of distribution fitting of a continuous distribution to Proban generated data.

In addition to these, it is possible to specify user defined distributions. How this is done is described in Section 3.9.3.

3.9.1 List of Distributions

The following table lists all distributions in Proban except the spline distribution (see the command CREATE VARIABLE ... DISTRIBUTION SPLINE-1DIM for an explanation of this). With each distribution is listed the input sequences, the parameters in each input sequence and the restrictions that apply to the parameters.

The distributions are documented in the SESAM User's Manual: Proban Distributions.

Distribution	Input sequence	Parameters
Beta	Mean-StD-Lim	Mean, Stand-Dev, Lower-Bound, Upper-Bound
	Mean-Cov-Lim	Mean, Coef-of-Var, Lower-Bound, Upper-Bound
	R-T-Lim	R, T, Lower-Bound, Upper-Bound
	R-S-Lim	R, S, Lower-Bound, Upper-Bound
	Low-MostL-Up	Lower-Bound, Most-Likely, Upper-Bound
		Lower-Bound < Mean < Upper-Bound, Mean*Coef-of-Var > 0
		Stand-Dev > 0, Coef-of-Var > 0, R > 0, S > 0, T > R
		Lower-Bound ≤ Most-Likely ≤ Upper-Bound
Binomial	N-Probab	Number, Probability
		Number ≥ 1 ,0< Probability< 1.
		Non-integer Number is replaced by nearest integer.
Burr	M-C-K-Low	M,C,K,Lower-Bound
		M>Lower-Bound, C>0, K>0
Chi-square	Mean-Low	Mean, Lower-Bound
	DoF-Low	Deg-of-Freed, Lower bound
		Mean > Lower-Bound, Deg-of-Freed > 0
Exponential	Mean-Low	Mean, Lower-Bound
	Rate-Low	Rate, Lower bound
		Mean > Lower-Bound, Rate > 0
Gamma	Mean-StD-Low	Mean, Stand-Dev, Lower-Bound
	Mean-Cov-Low	Mean, Coef-of-Var, Lower-Bound
	K-Lambda-Low	K, Lambda, Lower-Bound
		Mean > Lower-Bound, Mean*Coef-of-Var > 0

		Stand-Dev > 0, Coef-of-Var > 0, K > 0, Lambda > 0
Gen-Gamma	Al-B-C-Low	Alpha, B, C Lower-Bound
	Alpha > 0, C > 0	
Gen-Pareto	Sig-KsiP-Low	Sigma, KsiP, Low
		Sigma >0, |KsiP| > 0
Gumbel	Mean-StD	Mean, Stand-Dev
	Mean-Cov	Mean, Coef-of-Var
	Alpha	Alpha, B
		Mean*Coef-of-Var > 0
		Stand-Dev > 0, Coef-of-Var > 0, Alpha > 0
Hermit-Secon	Moments	Mean, Stand-Dev, Skewness, Kurtosis
	Central-Mom	Mean, Variance, Third-C-Mom, Fourth-C-Mom
		Stand-Dev > 0, Kurtosis > 0, Variance > 0, Fourth-C-Mom > 0
		(8/9)* Kurtosis ≥ Skewness*Skewness
Hermit-Trans	Moments	Mean, Stand-Dev, Skewness, Kurtosis
	Central-Mom	Mean, Variance, Third-C-Mom, Fourth-C-Mom
		Stand-Dev > 0, Kurtosis > 0, Variance > 0, Fourth-C-Mom > 0
		(8/9)* Kurtosis ≥ Skewness*Skewness
Inv-Gauss	Mean-StD-Low	Mean, Stand-Dev, Lower-Bound
	Mean-Cov-Low	Mean, Coef-of-Var, Lower-Bound
	Ksi-Lamb-Low	Ksi, Lambda, Lower-Bound
		Mean > Lower-Bound, Mean*Coef-of-Var > 0
		Stand-Dev > 0, Coef-of-Var > 0, Ksi > 0, Lambda > 0
Lognormal	Mean-StD-Low	Mean, Stand-Dev, Lower-Bound
	Mean-Cov-Low	Mean, Coef-of-Var, Lower-Bound
	Sigma-Mu-Low	Sigma, Mu, Lower-Bound
		Mean > Lower-Bound, Mean*Coef-of-Var > 0
		Stand-Dev > 0, Coef-of-Var > 0, Sigma > 0
Long-Higgins	NCycle-Delta	N-Cycles, Delta
		N-Cycles > 0, Delta > 0
Maxwell	Mean-Low	Mean, Lower-Bound
	Theta-Low	Theta, Lower bound

		Mean > Lower-Bound, Theta > 0
Multi-Normal	Cor-Std-Mean	Correlations (1,2), (1,3), ..., (2,3), ..., Stdv1, Stdv2, ..., Mean1, Mean2, ...
	Covar-Mean	Covariances (1,1), (1,2), ..., (2,2), (2,3), ..., (3,3), ..., Mean1, Mean2, ...
		The dimension (≤ 40) must be specified before the input sequence -1 < Correlation < 1, Stdv > 0, Covar(i,i) > 0 Covariance and correlation matrix must be positive definite
Normal	Mean-StD	Mean, Stand-Dev
	Mean-Cov	Mean, Coef-of-Var
		Mean*Coef-of-Var > 0
		Stand-Dev > 0, Coef-of-Var > 0
Onesi-Normal	Mean-Low	Mean, Lower-Bound
	Sigma-Low	Sigma, Lower bound
		Mean > Lower-Bound, Sigma > 0
Oval	Mean-Scale	Mean, Scale
		Scale > 0
Poisson	Mean	Mean
Rayleigh	Mean-Low	Mean, Lower-Bound
	Theta-Low	Theta, Lower bound
		Mean > Lower-Bound, Theta > 0
Student-t	Dof-Mean	Deg-of-Freed, Mean
	Deg-of-Freed > 0	
Triangle	Low-MostL-Up	Lower-Bound, Most-Likely, Upper-Bound
	Low-Mean-Up	Lower-Bound, Mean, Upper-Bound
		Lower-Bound £ Most-Likely £ Upper-Bound
		Mean must be within middle third of interval
Trunc-Normal	Mu-Sigma-Lim	Mu, Sigma, Lower-Bound, Upper-Bound
	Mu-Cov-Lim	Mu, Coef-of-Var, Lower-Bound, Upper-Bound
		Mu*Coef-of-Var > 0, Sigma > 0, Coef-of-Var > 0
		Lower-Bound < Upper-Bound
Uniform	Limits	Lower-Bound, Upper-Bound
	Mean-Low	Mean, Lower-Bound

	Mean-StD	Mean, Stand-Dev
	Mean-Cov	Mean, Coef-of-Var
		Stand-Dev > 0, Mean*Coef-of-Var > 0, Coef-of-Var > 0
		Lower-Bound < Upper-Bound, Mean > Lower-Bound
Weibull	Mean-StD-Low	Mean, Stand-Dev, Lower-Bound
	Mean-Cov-Low	Mean, Coef-of-Var, Lower-Bound
	Delt-Bet-Low	Delta, Beta, Lower-Bound
	Alp-Beta-Low	Alpha, Beta, Lower-Bound
		Mean > Lower-Bound, Mean*Coef-of-Var > 0, Beta > 0
		Stand-Dev > 0, Coef-of-Var > 0, Delta > 0, Alpha > 0

3.9.2 Distribution Fitting

A Proban distribution result can be fitted to distributions in the Proban Distributions Library. As an example, the distribution of NPV in Example 3.2 can be fitted to a normal distribution, the fitting parameters being Mean and COV. The input required to do this is

```
RUN DISTRIBUTION-ANALYSIS NPV
SET GRAPH HISTOGRAM COLUMNS 20
SET GRAPH HISTOGRAM FILLING HOLLOW
CREATE VARIABLE FitNPV ' ' FITTED-DISTRIBUTION Normal Mean-CoV FIT FIT
 RESULT LastAnalysis
```

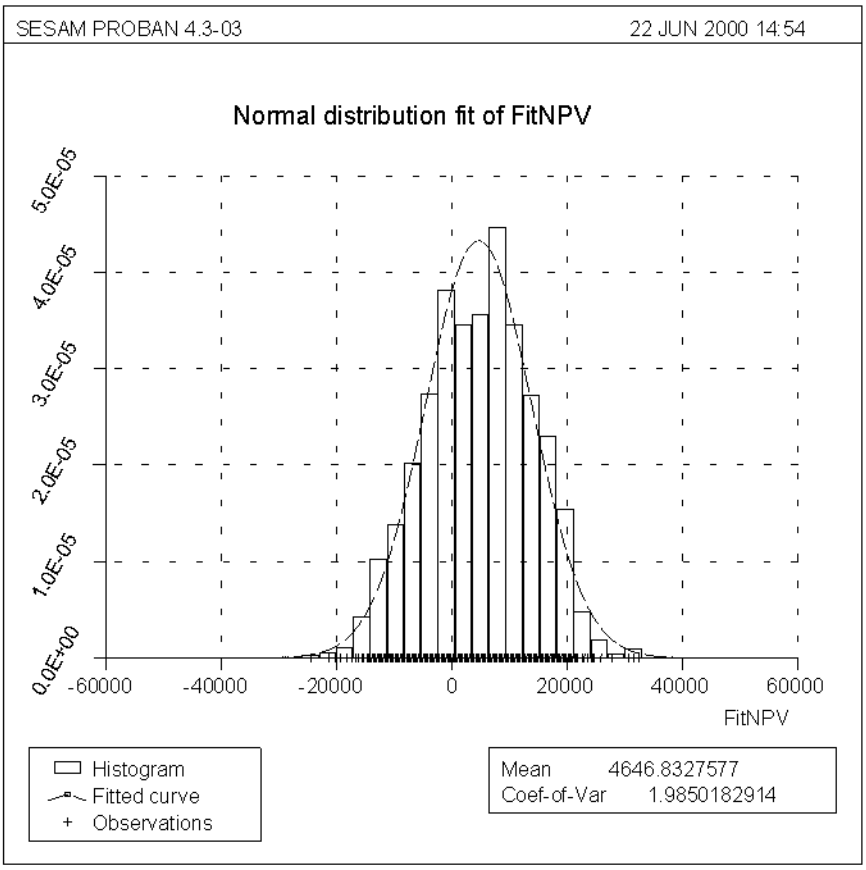


Figure 3.16 NPV fitted to normal distribution - Mean and COV

This input produces a fit that can be displayed as in Figure 3.16 by the following command:

```
DISPLAY FITTED-DISTRIBUTION FitNPV
```

The parameters are fitted to the result from a Proban distribution analysis by means of the Maximum Likelihood method. The vertical lines on the observation axis shows the density of the sampled observations. The histogram shows the contribution from each of the twenty intervals on the observation axis.

In the next example the distribution of a beta distributed random variable is calculated using a parameter study on a threshold value x. The result is fitted to the beta distribution

```
CREATE VARIABLE beta ' ' DISTRIBUTION Beta R-S-Lim 2 3 1 3
CREATE VARIABLE betax ' ' FUNCTION DIFFERENCE betax
DEFINE ANALYSIS-OPTION PARAMETER-STUDY ON
DEFINE PARAMETER-STUDY X ( ONLY GROUP 1.1 2.9 0.1 )
RUN PROBABILITY-ANALYSIS betax
CREATE VARIABLE beta_fit ' ' FITTED-DISTRIBUTION Beta R-S-Lim FITL1U4 FITL1U4
FIT0.5L0 FIT4.5U5 RESULT LastAnalysis
```

The resulting fit becomes

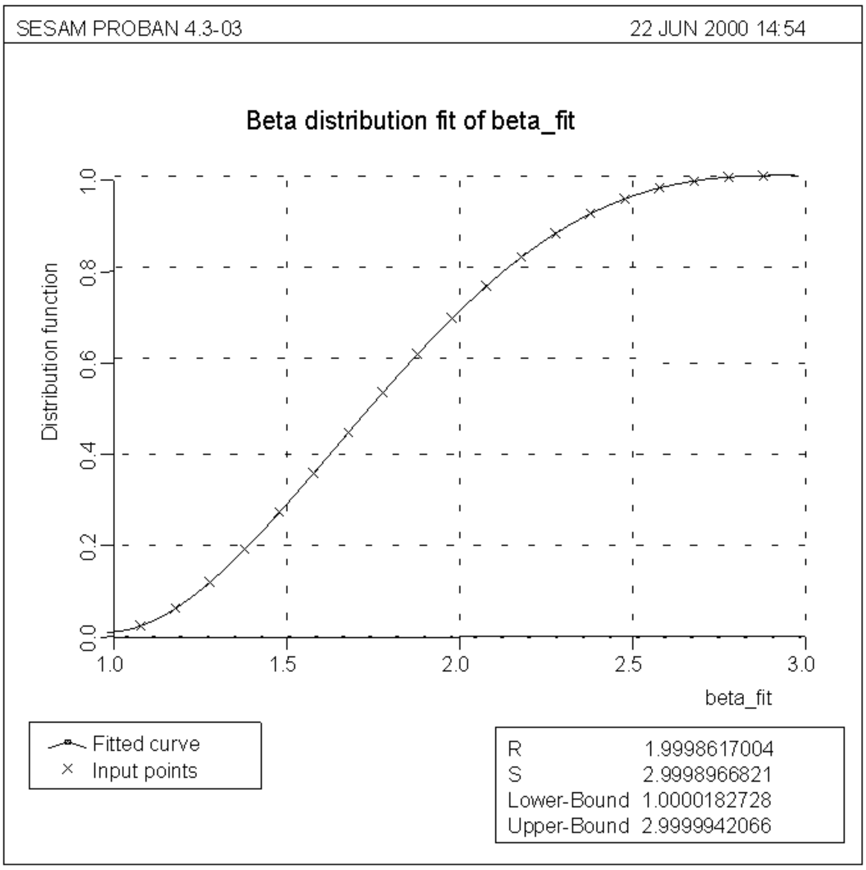


Figure 3.17 Fitting of beta distribution

By inspecting the input, we see that the fitted parameters are given starting point value and a lower or an upper bound. It is often necessary to specify an initial value and parameter bounds to the optimization algorithm in order to reach the (best) fit. The value after FIT is the starting point. The value after L is a lower bound on the parameter and the value after U is an upper bound.

Proban can also fit a distribution to the result of a parameter study on probability by means of the Least Squares method. For more details, see command CREATE VARIABLE ... FITTED-DISTRIBUTION.

3.9.3 User Defined Distributions

To add a user defined distribution to Proban requires that the distribution is programmed, and then linked into Proban.

Use the following sequence to add a user defined distribution:

1 Select a three character routine prefix for the distribution. This prefix should begin the name of each routine programmed with the distribution. For illustration of the process, assume that the chosen prefix is XXX. These prefixes cannot be used:

ATR BET BPM CDI CH2 CGR CHK CIQ COP CPM DDI DES DFU DIM DIS EXP EXT FOX
FU FX GAM GGM GUM HTM ICO IDI IIQ ING INI IPM IQ IQC LNM LOH LSC LSD
LSI MNR MOM MSG MXW NAM NMS NPM NRM NUM ONE OP OVA PAR PM PMI PMN PTZ
RAY SP1 SP2 STN STU TAC TOC TPA TRA TRI TRU TST UAT UNI USR VTZ WBL ZTV
ZVP

2 The distribution, input sequence and parameter(s) must be installed by modifying the routine USRINI. During start-up, Proban calls USRINI to install any user defined distributions. The delivered version of USRINI does not install any user defined distributions.

The distribution is allowed to have one input sequence. The input sequences that are in use in Proban already may be reused, or a new input sequence may be installed. The same applies to the parameters in the input sequence. If an existing input sequence or parameter is used, all restrictions that apply to the input sequence and parameters will also be in effect for the new distributions. These restrictions are described in the previous section.

Other details about the installation are described in USRINI itself. The location of USRINI is described in the installation guide. At the same place there is an example routine: USRINI.TST showing how the TST distribution is implemented.

3 Program the DDI routine for the distribution (e.g. named XXXDDI). This routine calculates the density function, distribution function and complementary distribution function from a fractile in the distribution. Proban is delivered with an example, called TSTDDI, that should be used as a template for the routine. The location of this routine is specified in the installation guide.

Proban requires good accuracy in the tail of the distribution, and may call the DDI routine with extreme tail values. Please be aware of this, and take special note of the possibility of an overflow (e.g. in the exp function) if a tail value is extreme.

4 The DDI routine is activated through the routine USRDDI. USRDDI must be modified by inserting a call to the DDI routine for the distribution. See the documentation in USRDDI itself for further clarification. Proban is delivered with a USRDDI routine that does not call any user defined distributions. The location of USRDDI is described in the installation guide. At the same place there is an example routine: USRDDI.TST showing how the TST distribution is implemented.

5 Proban is delivered with an object library, called USER. The location of the library is described in the installation guide. This library contains the user defined distributions (it is delivered with only USRINI and USRDDI). Take a copy of this library. Then compile USRINI, USRDDI and the distribution DDI routine (e.g. XXXDDI) (and possibly other routines that are needed by the new DDI routine) and place the object codes in the USER library.

6 ink the USER library into Proban using the link command file or makefile delivered with Proban The procedure for doing this is installation dependent and is described in the installation guide.

7 Check the distribution by use of the PRINT DISTRIBUTION command. The HIGH-RESOLUTION print option will print warnings if the DDI routine seems to give wrong results. Also try giving some extreme tail values using the FRACTILE and PROBABILITY options.

3.10 Model Functions

The library of model functions are divided into sublibraries (also called function libraries). This subdivision is used in order to group functions into logically coherent groups, and to be able to mask off temporarily some of the functions (see the command SELECT FUNCTION-LIBRARY).

All functions and sublibraries are named, and are referenced by name in Proban.

At the top of the hierarchy resides a routine, that must have the name FUNCLB. This routine is called by Proban when it needs information from a model function or sublibrary. FUNCLB then calls a number of sublibrary routines, and each of these controls a number of model functions.

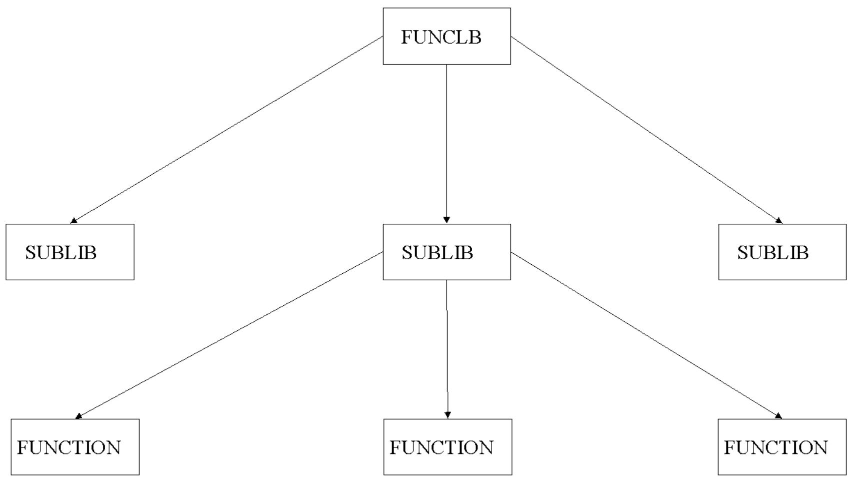


Figure 3.18 Organisation of the function library

In addition to this function library tree, Proban has some built-in function libraries, and is able to use the Proban Version 2 function library LIBLIM. Also, simple functions can be created on input.

The built in libraries are described in Section 3.10.1 and the compatibility issues regarding Proban Version 3 are described in Section 3.10.4. Section 3.10.2 shows an example of how to create a function formula on input.

Proban is delivered with a library that contains the examples from the example manual in one sublibrary called: “Examples”. This library is separated into several FORTRAN routines. The location of these routines is described in the installation guide.

Section 3.10.3 describes how to create a private function library and add model functions to this.

The contents of a sublibrary may be printed using the PRINT FUNCTION LIBRARY command. The description of each function may be printed using the PRINT FUNCTION DESCRIPTION command.

3.10.1 The Built-in Function Libraries

Proban Version 4-3 contains three built-in function libraries, with the names “Misc”, “Math” and “Prob-Logical”.

The “Math” library contains a large number of basic mathematical functions, the “Prob-Logical” library contains probability functions and logical functions and the “Misc” library contains some functions that are generally useful.

These routines are useful building blocks, from which many model functions can be built.

The following is a list of the print of the contents of the three libraries. NumArg is the number of arguments in the function. If the number of arguments is specified as: Input, it means that the function does not have a fixed number of arguments. Examples of this are the Sum and the Product functions.

```
+---------------------------------------+
!               Sublibrary              !
+---------------------------------------+
!                  Math                 !
! Miscellaneous mathematical functions  !
+---------------------------------------+
```

```
Function      Dimen  NArg NOp Description
--------------------------------------------------------------------------------
Abs               1     1   0 Absolute value
ArcCos-Deg        1     1   0 ArcCosinus, returning a value in [0,180]
ArcCos-Rad        1     1   0 ArcCosinus, returning a value in [0,pi]
ArcSin-Deg        1     1   0 ArcSinus, returning a value in [-90,90]
ArcSin-Rad        1     1   0 ArcSinus, returning a value in [-pi/2,pi/2]
ArcTan-Deg        1     1   0 ArcTangens, returning a value in ]-90,90[
ArcTan-Rad        1     1   0 ArcTangens, returning a value in ]-pi/2,pi/2[
Cos-Degrees       1     1   0 Cosinus of an argument in degrees (0-360)
Cos-Radians       1     1   0 Cosinus of an argument in radians
Cosh              1     1   0 Hyperbolic cosinus: (exp(x)+exp(-x))/2
Exp               1     1   0 Exponential function
Fraction          1     1   0 Fraction part of a number
Indicator-EQ      1     2   0 Indicator:  = 1 if X1 = X2,  = 0 otherwise
Indicator-GE      1     2   0 Indicator:  = 1 if X1 >= X2,  = 0 otherwise
Indicator-GT      1     2   0 Indicator:  = 1 if X1 > X2,  = 0 otherwise
Indicator-LE      1     2   0 Indicator:  = 1 if X1 <= X2,  = 0 otherwise
Indicator-LT      1     2   0 Indicator:  = 1 if X1 < X2,  = 0 otherwise
Integer           1     1   0 Strip away decimal part of a number
Log               1     1   0 Natural logarithm
Log10             1     1   0 Logarithm with base 10
Power             1     2   0 Power function: X1**X2
Round             1     1   0 Nearest integer to a number (0.5 -> 1, -0.5 -> -1)
Sign              1     1   0 The sign of a number, or 0 if it is = 0
Sin-Degrees       1     1   0 Sinus of an argument in degrees (0-360)
Sin-Radians       1     1   0 Sinus of an argument in radians
Sinh              1     1   0 Hyperbolic sinus: (exp(x)-exp(-x))/2
Sqrt              1     1   0 Square root
Square            1     1   0 Square of a value
```

```
Tan-Degrees       1     1   0 Tangent of an argument in degrees (0-360)
Tan-Radians       1     1   0 Tangent of an argument in radians
Tanh              1     1   0 Hyperbolic tangent


                  +-------------------------------------------+
                  !                Sublibrary                 !
                  +-------------------------------------------+
                  !               Prob-Logical                !
                  ! Probability functions for logical systems !
                  +-------------------------------------------+

Function     Dimen  NArg NOp Description
--------------------------------------------------------------------------------
EqProb-AND        1     2   0 AND gate: Prob = p**n
EqProb-EQV        1     2   0 EQV gate: Prob = p**n+(1-p)**n
EqProb-NEQV       1     2   0 NEQV gate: Prob = 1-p**n-(1-p)**n
EqProb-OR         1     2   0 OR gate: Prob = 1-(1-p)**n
EqRelIn-AND       1     2   0 AND gate: B = -InvPHI( PHI(-B)**N )
EqRelIn-EQV       1     2   0 EQV gate: PHI(-F) = PHI(-B)**N+PHI(B)**N
EqRelIn-NEQV      1     2   0 NEQV gate: PHI(-F) = 1-PHI(-B)**N-PHI(B)**N
EqRelIn-OR        1     2   0 OR gate: B = InvPHI( PHI(B)**N )
Prob-AND          1 Input   0 AND gate: Prob = p1*p2*...*pn
Prob-EQV          1 Input   0 EQV gate: Prob=(1-p1)*...*(1-pn)+p1*...*pn
Prob-NEQV         1 Input   0 NEQV gate: Prob = 1-(1-p1)*...*(1-pn)-p1*...*pn
Prob-NOT          1     1   0 NOT gate: Prob = 1-p
Prob-OR           1 Input   0 OR gate: Prob = 1-(1-p1)*(1-p2)*...*(1-pn)
RelIn-AND         1 Input   0 AND gate: B = -InvPhi( PHI(-B1)*...PHI(-BN) )
RelIn-EQV         1 Input   0 EQV gate: PHI(-F) = PROD PHI(-Bi)+PROD PHI(Bi)
RelIn-NEQV        1 Input   0 NEQV gate: PHI(-F) = 1-PROD PHI(-Bi)- PROD PHI(Bi)
RelIn-NOT         1     1   0 NOT gate: Reliability Index = -B
RelIn-OR          1 Input   0 OR gate: B = InvPHI( PHI(B1)*...PHI(BN) )


                              +--------------------+
                              !    Sublibrary      !
                              +--------------------+
                              !    Special-Fu      !
                              ! Special functions  !
                              +--------------------+

Function     Dimen  NArg NOp Description
--------------------------------------------------------------------------------
ErrFun            1     1   0 Error Function
ErrCFun           1     1   0 Complementary error function
Gamma             1     1   0 Gamma Function
LnGamma           1     1   0 Logarithm of Gamma function
```

```
                    +-------------------------------------+
                    !             Sublibrary              !
                    +-------------------------------------+
                    !            Distribution             !
                    ! Functions related to distributions !
                    +-------------------------------------+

Function     Dimen  NArg NOp Description
-------------------------------------------------------------------------------
Den-Beta         1     5   0 Beta distribution, Density function
Den-Burr         1     5   0 Burr distribution, Density function
Den-Chi-squa     1     3   0 Chi-square distribution, Density function
Den-Exponent     1     3   0 Exponential distribution, Density function
Den-Gamma        1     4   0 Gamma distribution, Density function
Den-Gen-Gamm     1     5   0 Gen-Gamma distribution, Density function
Den-Gumbel       1     3   0 Gumbel distribution, Density function
Den-Hermit-s     1     5   0 Hermit-secon distribution, Density function
Den-Hermit-t     1     5   0 Hermit-trans distribution, Density function
Den-Inv-Gaus     1     4   0 Inv-Gauss distribution, Density function
Den-Lognorma     1     4   0 Lognormal distribution, Density function
Den-Long-Hig     1     3   0 Long-Higgins distribution, Density function
Den-Maxwell      1     3   0 Maxwell distribution, Density function
Den-Normal       1     3   0 Normal distribution, Density function
Den-Onesi-No     1     3   0 Onesi-Normal distribution, Density function
Den-Oval         1     3   0 Oval distribution, Density function
Den-Rayleigh     1     3   0 Rayleigh distribution, Density function
Den-Student-     1     3   0 Student-t distribution, Density function
Den-Triangle     1     4   0 Triangle distribution, Density function
Den-Trunc-No     1     5   0 Trunc-Normal distribution, Density function
Den-Weibull      1     4   0 Weibull distribution, Density function
Dis-Beta         1     5   0 Beta distribution, Distribution function
Dis-Burr         1     5   0 Burr distribution, Distribution function
Dis-Chi-squa     1     3   0 Chi-square distribution, Distribution function
Dis-Exponent     1     3   0 Exponential distribution, Distribution function
Dis-Gamma        1     4   0 Gamma distribution, Distribution function
Dis-Gen-Gamm     1     5   0 Gen-Gamma distribution, Distribution function
Dis-Gumbel       1     3   0 Gumbel distribution, Distribution function
Dis-Hermit-s     1     5   0 Hermit-secon distribution, Distribution function
Dis-Hermit-t     1     5   0 Hermit-trans distribution, Distribution function
Dis-Inv-Gaus     1     4   0 Inv-Gauss distribution, Distribution function
Dis-Lognorma     1     4   0 Lognormal distribution, Distribution function
Dis-Long-Hig     1     3   0 Long-Higgins distribution, Distribution function
Dis-Maxwell      1     3   0 Maxwell distribution, Distribution function
Dis-Normal       1     3   0 Normal distribution, Distribution function
Dis-Onesi-No     1     3   0 Onesi-Normal distribution, Distribution function
Dis-Oval         1     3   0 Oval distribution, Distribution function
Dis-Rayleigh     1     3   0 Rayleigh distribution, Distribution function
Dis-Student-     1     3   0 Student-t distribution, Distribution function
Dis-Triangle     1     4   0 Triangle distribution, Distribution function
Dis-Trunc-No     1     5   0 Trunc-Normal distribution, Distribution function
Dis-Weibull      1     4   0 Weibull distribution, Distribution function
Inv-Beta         1     5   0 Beta distribution, Inverse distribution fct.
```

```
Inv-Burr        1     5   0 Burr distribution, Inverse distribution fct.
Inv-Chi-squa    1     3   0 Chi-square distribution, Inverse distribution fct.
Inv-Exponent    1     3   0 Exponential distribution, Inverse distribution fct
Inv-Gamma       1     4   0 Gamma distribution, Inverse distribution fct.
Inv-Gen-Gamm    1     5   0 Gen-Gamma distribution, Inverse distribution fct.
Inv-Gumbel      1     3   0 Gumbel distribution, Inverse distribution fct.
Inv-Hermit-s    1     5   0 Hermit-secon distribution, Inverse distribution fc
Inv-Hermit-t    1     5   0 Hermit-trans distribution, Inverse distribution fc
Inv-Inv-Gaus    1     4   0 Inv-Gauss distribution, Inverse distribution fct.
Inv-Lognorma    1     4   0 Lognormal distribution, Inverse distribution fct.
Inv-Long-Hig    1     3   0 Long-Higgins distribution, Inverse distribution fc
Inv-Maxwell     1     3   0 Maxwell distribution, Inverse distribution fct.
Inv-Normal      1     3   0 Normal distribution, Inverse distribution fct.
Inv-Onesi-No    1     3   0 Onesi-Normal distribution, Inverse distribution fc
Inv-Oval        1     3   0 Oval distribution, Inverse distribution fct.
Inv-Rayleigh    1     3   0 Rayleigh distribution, Inverse distribution fct.
Inv-Student-    1     3   0 Student-t distribution, Inverse distribution fct.


Function     Dimen  NArg NOp Description
-------------------------------------------------------------------------------
Inv-Triangle    1     4   0 Triangle distribution, Inverse distribution fct.
Inv-Trunc-No    1     5   0 Trunc-Normal distribution, Inverse distribution fc
Inv-Weibull     1     4   0 Weibull distribution, Inverse distribution fct.
UtX-Beta        1     5   0 Beta distribution, Inverse of Std. Normal
UtX-Burr        1     5   0 Burr distribution, Inverse of Std. Normal
UtX-Chi-squa    1     3   0 Chi-square distribution, Inverse of Std. Normal
UtX-Exponent    1     3   0 Exponential distribution, Inverse of Std. Normal
UtX-Gamma       1     4   0 Gamma distribution, Inverse of Std. Normal
UtX-Gen-Gamm    1     5   0 Gen-Gamma distribution, Inverse of Std. Normal
UtX-Gumbel      1     3   0 Gumbel distribution, Inverse of Std. Normal
UtX-Hermit-s    1     5   0 Hermit-secon distribution, Inverse of Std. Normal
UtX-Hermit-t    1     5   0 Hermit-trans distribution, Inverse of Std. Normal
UtX-Inv-Gaus    1     4   0 Inv-Gauss distribution, Inverse of Std. Normal
UtX-Lognorma    1     4   0 Lognormal distribution, Inverse of Std. Normal
UtX-Long-Hig    1     3   0 Long-Higgins distribution, Inverse of Std. Normal
UtX-Maxwell     1     3   0 Maxwell distribution, Inverse of Std. Normal
UtX-Onesi-No    1     3   0 Onesi-Normal distribution, Inverse of Std. Normal
UtX-Oval        1     3   0 Oval distribution, Inverse of Std. Normal
UtX-Rayleigh    1     3   0 Rayleigh distribution, Inverse of Std. Normal
UtX-Student-    1     3   0 Student-t distribution, Inverse of Std. Normal
UtX-Triangle    1     4   0 Triangle distribution, Inverse of Std. Normal
UtX-Trunc-No    1     5   0 Trunc-Normal distribution, Inverse of Std. Normal
UtX-Weibull     1     4   0 Weibull distribution, Inverse of Std. Normal
XtU-Beta        1     5   0 Beta distribution, Standard Normal fractile
XtU-Burr        1     5   0 Burr distribution, Standard Normal fractile
XtU-Chi-squa    1     3   0 Chi-square distribution, Standard Normal fractile
XtU-Exponent    1     3   0 Exponential distribution, Standard Normal fractile
XtU-Gamma       1     4   0 Gamma distribution, Standard Normal fractile
XtU-Gen-Gamm    1     5   0 Gen-Gamma distribution, Standard Normal fractile
XtU-Gumbel      1     3   0 Gumbel distribution, Standard Normal fractile
XtU-Hermit-s    1     5   0 Hermit-secon distribution, Standard Normal fractil
```

```
XtU-Hermit-t     1      5   0 Hermit-trans distribution, Standard Normal fractil
XtU-Inv-Gaus     1      4   0 Inv-Gauss distribution, Standard Normal fractile
XtU-Lognorma     1      4   0 Lognormal distribution, Standard Normal fractile
XtU-Long-Hig     1      3   0 Long-Higgins distribution, Standard Normal fractil
XtU-Maxwell      1      3   0 Maxwell distribution, Standard Normal fractile
XtU-Onesi-No     1      3   0 Onesi-Normal distribution, Standard Normal fractil
XtU-Oval         1      3   0 Oval distribution, Standard Normal fractile
XtU-Rayleigh     1      3   0 Rayleigh distribution, Standard Normal fractile
XtU-Student-     1      3   0 Student-t distribution, Standard Normal fractile
XtU-Triangle     1      4   0 Triangle distribution, Standard Normal fractile
XtU-Trunc-No     1      5   0 Trunc-Normal distribution, Standard Normal fractil
XtU-Weibull      1      4   0 Weibull distribution, Standard Normal fractile


                      +----------------------------------+
                      !            Sublibrary            !
                      +----------------------------------+
                      !               Misc               !
                      ! Miscellaneous general functions !
                      +----------------------------------+

Function      Dimen  NArg NOp Description
-------------------------------------------------------------------------------
Difference       1      2   0 Difference X1 - X2
Division         1      2   0 Division X1 / X2
Identity         1      1   0 Identity: f(x) = x
Linear-Comb      1 Input   0 Linear combination: x1*x2 + x3*x4 + ...
Log-Diff         1      2   0 Difference: Log(X1) - Log(X2)
Maximum          1 Input   0 Maximum of any number of variables
Minimum          1 Input   0 Minimum of any number of variables
Polynom-1        1      4   0 Polynomium of degree 1
Polynom-2        1      5   0 Polynomium of degree 2
Polynom-3        1      6   0 Polynomium of degree 3
Polynom-4        1      7   0 Polynomium of degree 4
Polynom-N        1 Input   0 Polynomium(N,X,X0,C0,...): Sum of Ci*((X-X0)**i)
Power-Diff       1      3   0 Difference: X1**X3 - X2**X3
Product          1 Input   0 Product of any number of variables
Sequence         2      2   1 F(i) = X(i): Arguments -> Vector
SignLogDiff      1      2   0 l(X1)-l(X2): l(X)=-1-log(-X),X,1+log(X);<=-1,,>=1
SignPowDiff      1      3   0 Sign(X1)*(Abs(X1)**X3) - Sign(X2)*(Abs(X2)**X3)
Sum              1 Input   0 Sum of any number of variables
```

3.10.2 Create Function Formula Interactively

Functions can be modelled on input by using the command CREATE FUNCTION FORMULA and CHANGE FUNCTION FORMULA. As an example consider the beam example in the example manual /3/. Rather than programming the formula and linking it into Proban the function formula can be created on input as shown below. The syntax is described under the command CREATE FUNCTION FORMULA. However, notice that the order of calculation is according to the FORTRAN syntax.

```
CREATE FUNCTION LoadPart 'Load part of moment and shear' FORMULA
                 ( ONLY P1       'Applied load 1'
                        P2       'Applied load 2'
                        L1       'Location of load 1'
                        L2       'Location of load 2'
                        Span     'Beam span'
                 )
                   '(P1*(1-L1/Span)+P2*L2/Span)'

CREATE FUNCTION MomForml 'Moment at end of beam' FORMULA
                 ( ONLY P1       'Applied load at end position'
                        P2       'Applied load at other position'
                        L1       'Location load at end position'
                        L2       'Location load at other position'
                        Span     'Beam span'
                        Depth    'Effective Depth'
                        Ts       'Steel yield stress'
                        As       'Steel area'
                        K        'Stress-strain coefficient'
                        Width    'Width of beam'
                        Tc       'Concrete compressive strength'
                 )
                 (      'As*Depth*Ts - K*As**2*Ts**2/Width/Tc'
                        '- L1*LoadPart(P1,P2,L1,L2,Span)'
                 )

CREATE FUNCTION ShrForml 'Shear at end of beam' FORMULA

                 ( ONLY P1       'Applied load at end position'
                        P2       'Applied load at other position'
                        L1       'Location load at end position'
                        L2       'Location load at other position'
                        Span     'Beam span'
                        Depth    'Effective Depth'
                        Ts       'Steel yield stress'
                        Width    'Width of beam'
                        Tc       'Concrete compressive strength'
                        Av       'Shear steel area'
                        Spacing  'Shear steel spacing'
                 )
                 (      '0.2*Sqrt(Tc)*Width*Depth'
                        '+ Av*Depth*Ts/Spacing'
                        '- LoadPart(P1,P2,L1,L2,Span)'
                 )

CREATE VARIABLE
LOOP
  P1       'Applied load 1'                DISTR Normal Mean-StD 28000. 8400.
  P2       'Applied load 2'                DISTR Normal Mean-StD 28000. 8400.
  L1       'Location load 1'               DISTR Normal Mean-StD 750. 60.
  L2       'Location load 2'               DISTR Normal Mean-StD 750. 60.
  Depth    'Effective Depth'               DISTR Normal Mean-StD 300. 15.
```

```
  Ts       'Steel yield stress'            DISTR Normal Mean-StD 360. 36.
  As       'Steel area'                    DISTR Normal Mean-StD 452. 22.6
  K        'Stress-strain coefficient'     DISTR Normal Mean-StD 0.55 0.055
  Width    'Width of beam'                 DISTR Normal Mean-StD 120. 6.
  Tc       'Concrete compressive strength' DISTR Normal Mean-StD 40. 6.
  Av       'Shear steel area'              DISTR Normal Mean-StD 35. 1.75
  Spacing  'Shear steel spacing'           DISTR Normal Mean-StD 300. 45.
  Span     'Beam span'                     FIXED 3000.
  Moment1  'Moment limit state at 1'       FUNCTION  MomForml
            P1 P2 L1 L2 Span Depth Ts As K Width Tc
  Moment2  'Moment limit state at 2'       FUNCTION  MomForml
            P2 P1 L2 L1 Span Depth Ts As K Width Tc
  Shear0   'Shear limit state at 0'         FUNCTION  ShrForml
            P1 P2 L1 L2 Span Depth Ts Width Tc Av Spacing
  Shear3   'Shear limit state at 3'         FUNCTION  ShrForml
            P2 P1 L2 L1 Span Depth Ts Width Tc Av Spacing
END

PRINT FUNCTION DESCRIPTION LoadPart
PRINT FUNCTION FORMULA     MomForml
```

As can be seen from the input the moment formula and the shear formula have a common load part. This load part is created separately as a function formula. The load part formula is then used in the definition of the moment formula and the shear formula. Any one-dimensional function in any (user defined) function library and any formula created on input can be used as a reference in the definition of a function formula. The only limitation is that a function formula cannot (in)directly reference itself.

Notice that it is possible and often useful to divide a function formula into a number of smaller formulas.

The command PRINT FUNCTION DESCRIPTION LoadPart produces:

```
                       +--------------------------------+
                       !            Function            !
                       +--------------------------------+
                       !            LoadPart            !
                       ! Load part of moment and shear !
                       +--------------------------------+

                 The function belongs to sublibrary: SYMBOLIC


                    Gradients must be calculated numerically


             Name         Description
----------------------------------------------------------------------------
Arguments:   P1           Applied load 1
             P2           Applied load 2
             L1           Location of load 1
             L2           Location of load 2
             Span         Beam span
```

```
Formula:     (P1*(1-L1/Span)+P2*L2/Span)
```

The formula is printed at the end of the function description.

The command PRINT FUNCTION FORMULA MomForml produces:

```
                           +------------------------+
                           !        Function        !
                           +------------------------+
                           !        MomForml        !
                           ! Moment at end of beam  !
                           +------------------------+

                  Gradients must be calculated numerically

Name         Description                                         Value Index
----------------------------------------------------------------------------
P1           Applied load at end position                        V1
P2           Applied load at other position                      V2
L1           Location load at end position                       V3
L2           Location load at other position                     V4
Span         Beam span                                           V5
Depth        Effective Depth                                     V6
Ts           Steel yield stress                                  V7
As           Steel area                                          V8
K            Stress-strain coefficient                           V9
Width        Width of beam                                       V10

Tc           Concrete compressive strength                       V11


                  Formula Interpretation                        SUB PAGE:     2

NOMENCLATURE:

Operator    +,-,*,/,**,Function Name
Operands    Positions of Operand Values
Result      Position of Resulting Value


Operator     Operands                                            Result
----------------------------------------------------------------------------

*            V8 V6                                               V12
*            V12 V7                                              V13
**           V8 2.0                                              V14
*            V9 V14                                              V15
**           V7 2.0                                              V16
/            V16 V10                                             V17
/            V17 V11                                             V18
```

```
*           V15 V18                                            V19
-           V13 V19                                            V20
LoadPart    V1 V2 V3 V4 V5                                     V21

*           V3 V21                                             V22
-           V20 V22                                            V23

Formula:    As*Depth*Ts - K*As**2*Ts**2/Width/Tc
            - L1*LoadPart(P1,P2,L1,L2,Span)
```

the print is of the function arguments, the order of calculation and the input formula text. The order calculation has the arguments first and the function value last and provides an additional means to verify the formula.

Values of function options for a function referenced by a function formula can be entered in the argument list for that function. The function options applied to a function referenced in a function formula are its current default options overwritten by options entered in the argument list. If a function with dimension defined by a function option is to be referenced, then its dimension must be set to one prior to the creation of the function formula, as shown below:

```
ASSIGN FUNCTION-OPTION FUNCTION FuncOptTest Opt-5-Menu SumTerm
ASSIGN FUNCTION-OPTION FUNCTION FuncOptTest Opt-7-NCoord 1
CREATE FUNCTION FUNSYM 'Symbolic Function involving options' FORMULA
   ( ONLY
     A 'Arg A'
     B 'Arg B'
     C 'Arg C' )
   ( 'A+B+C+FUNCOPTTEST(Opt_1_Text=''ab.'''')c'','
     'Opt_2_intege=2,' 'Opt_3_double=4.2,' 'Opt_5_menu=''SumTerm'','
     'Opt_6_Narg=3,A,2*B,3*C)' )
```

The usage of apostrophes in the input file above gives text value of Opt_1_Text=ab.')c

3.10.3 Creating and Updating a Private Function Library

The routines that must be programmed should be kept in one directory. It is also recommended to compile all the routines and keep the object code in an object library, which is then linked into Proban. Proban is delivered with tools that facilitate maintenance of the object library (a Makefile on Unix - See the installation guide for the location of this file). These files are self-explanatory - please read the comments in the files.

To get started, take a copy of the function library (FUNCLB) that is delivered with Proban and of the source code that is delivered with it, as well as the file that is used to maintain the object library.

This function library contains a dummy LIBLIM routine as well as the example sublibrary.

If you need to incorporate an existing LIBLIM, remove the LIBLIM delivered with Proban and simply compile your own LIBLIM routine(s) (including all sublibraries and functions) and add them to the FUNCLB

object library. If you do not need to program a new function, you are ready to link Proban (see step 5) below).

The example library may be removed by editing the call to EXAMLB out of FUNCLB. Remember to change the number of sublibraries in FUNCLB.

To add a function to the function library, follow the procedure described here. The location of the templates is described in the installation guide. The templates contain much documentation that will not be mentioned here.

1 Program the function (using FORTRAN 90). There is a number of templates available for different types of functions. The complexity of the function is dependent on the capabilities of the function. Use the template that fits the functions capabilities in order to avoid unnecessary work:

   FUNC10.DOC is used for a function that returns one value and does not calculate derivatives.

   FUNC11.DOC is used for a function that returns one value and provides first order derivatives.

   FUNC12.DOC is used for a function that returns one value and provides first and second order derivatives.

   FUNCN0.DOC is used for a function that returns a vector value and does not calculate derivatives.

   FUNCN1.DOC is used for a function that returns a vector value and provides first order derivatives.

   FUNCN2.DOC is used for a function that returns a vector value and provides first and second order derivatives.

2 Insert call to the function into a sublibrary. It may be necessary or desirable to create a new sublibrary first. Use the template SUBLIB.DOC or a copy of an existing sublibrary routine to do this. Remember to change the value specifying the number of functions in the sublibrary.

3 Add a call to the new sublibrary to FUNCLB, if a new sublibrary was created. If not, this step can be skipped. Remember to change value specifying the number of sublibraries in FUNCLB.

4 Compile all new and modified routines, and update the object library with the object modules.

5 Link Proban using the link command procedure or makefile that is delivered with Proban. Specify the location of your private function object library in the command.

6 Check the function value and gradients by use of the PRINT FUNCTION VALUE and PRINT FUNCTION GRADIENT commands. It is important that the function value and especially the gradients are somehow checked.

When programming model functions, it is usually a good idea to separate each part of the model into different functions in order to gain more flexibility in the modelling and analysis.

As an example, consider the model function:

$$\int_a^b \frac{1}{K(x, \boldsymbol{y})} dx - Ct = f(a, b, \boldsymbol{y}) - g(C, t)$$

The immediate approach is to code the difference f-g as one function. However, it is much better to code f as a function in itself, and model g using the already available Product function.

Modelling f and g separately gives the following advantages:

- It is easy to reformulate the problem, e.g. to log(f) - log(g) (using the Log-Diff function instead of the Difference function). If f - g had been coded as one function, such a remodelling would require reprogramming and subsequent re-linking.
- It becomes possible to examine the behaviour of and g separately, e.g. look at their distributions.
- The individual functions may be reused in other modelling situations.

3.10.4 Compatibility with Proban Version 2 - LIBLIM

Proban can use the existing LIBLIM routines without any changes. However, it is not possible to use the new facilities without converting the function to the new format.

The only slight conversion problem is, that the names of function arguments will be truncated from 25 to 12 characters, and hyphens (-) are inserted instead of blank spaces in the names between words. This may cause some function argument names to be identical inside the same function. This may again give problems when assigning sensitivity calculation or a parameter study to such parameters, because Proban cannot distinguish between the different arguments. The chance of this being a problem is in reality very small.

The function names from Proban Version 2 are unchanged, except that blank spaces between words in the names are substituted with hyphens (-) (e.g. “Func 1 is” changed to “Func-1”).

The sublibraries under LIBLIM will be named LIBLIM-1, LIBLIM-2 etc.

Proban needs both a FUNCLB and a LIBLIM routine in order to run. It needs to call both in order to be able to provide the compatibility to Proban Version 2. Thus, if only LIBLIM is used, a dummy version of FUNCLB must be linked in, and vice versa. Proban is delivered with a dummy version of LIBLIM that has no sublibraries. Users that do not need LIBLIM can simply use this, while users that have their own LIBLIM will need to substitute their LIBLIM with the one provided.

It is possible to mix old LIBLIM routines with routines that are programmed in the new format. The two hierarchies of functions are completely separate, and the routines are simply placed in the same object library.

3.11 Various Hints

This section contains various hints on how to facilitate the use of Proban.

3.11.1 Importing Plot Files into Documents

Proban will orient plots along the long edge of the paper. Thus, if a plot produced by Proban is imported into a document and is intended to be presented with text (as in this manual), it will most likely be oriented in the wrong direction.

Some word processors cannot rotate such a picture. If you have this problem, use the following procedure instead:

1 Write the plot file in SESAM-NEUTRAL format.

2 Use the program PLTCNV_EXT, which is delivered with Proban to convert it to another format. The input to PLTCNV_EXT will be:

```
SCALE = 0.9
OUTPUT-FILE-NAME=<the proper file name>
<input_file>.PLO
<output_format>
EXIT
```

The scale command is necessary for conversion to Postscript files, but may not be needed otherwise. For the list of proper output formats, run PLTCNV_EXT interactively (Postscript is PSCR).

For documents maintained on a PC, the CGM or HPGL-7550 format may be more suitable than Postscript (the latter format is e.g. recognised by MS-Word when renamed to have a .HGL suffix). However, if such a file is written when running VMS, it cannot be imported directly into a PC document because of file format differences between VMS and DOS. In this case, it is better to write a file in SESAM-NEUTRAL format, and then use PLTCNV_EXT to convert it (as above), using HP70 as the output format. During this conversion, the SCALE command is not needed, and NO-ROTATE should be used instead.

Please note that it is necessary to write one plot only to each plot file that is to be imported into a word processor.

3.11.2 If the Required Plot Format is not Available

If the plot format required by your printer/plotter is not available in the SET PLOT FORMAT command, you can try the following:

1 Write the plot file in SESAM-NEUTRAL format.

2 Use the program PLTCNV_EXT, which is delivered with Proban to convert it to another format.

This program includes several formats that are not available in Proban. However, the extra formats are not tested and supported as well as the formats included in Proban itself.

3.11.3 Problems with Convergence During FORM/SORM Analysis

In some cases the calculation of reliability index using FORM or SORM fails. There are basically two things that can go wrong:

1 The search for the design point(s) fails.

In this case, Proban will display a message stating that the linearisation of the design point has failed. The Kuhn-Tucker convergence criterion (KTO) can be monitored by using the commands

DEFINE ANALYSIS-OPTION INTERMEDIATE-PRINT LEVEL ...

DEFINE ANALYSIS-OPTION NESTED-ANALYSIS INTERMEDIATE-PRINT GLOBAL ...

DEFINE ANALYSIS-OPTION NESTED-ANALYSIS INTERMEDIATE-PRINT SYSTEM ...

and inspect the development of the KTO. Very often it converges initially to a small value and then do not get further. This is most often caused by lack of numerical precision in the calculation of functions and especially in the calculation of gradients. A possible remedy is to change the KTO to a larger value, using the command:

DEFINE FORM-SORM OPTIMIZATION

DEFINE FORM-SORM NESTED-ANALYSIS GLOBAL ...

DEFINE FORM-SORM NESTED-ANALYSIS SYSTEM ...

It may also be that the precision of a numeric derivative is poor. Differentiation increments can be adjusted by using the commands

DEFINE ANALYSIS-OPTION DIFFERENTIATION

DEFINE ANALYSIS-OPTION NESTED-ANALYSIS DIFFERENTIATION

DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION DIFFERENTIATION

Analytic derivatives can be coded together with the corresponding function and be linked into the program in order to increase numerical precision in the derivatives and also to reduce computational work.

In other cases the design point search does not find a path leading to the target. A remedy is to set starting point and optimization bounds for selected variables in order to restrict the search. This is particularly useful in connection with nested reliability analyses where ill-conditioned inner loop calculations may arise if the outer loop optimization variables are unrestricted. This is done by using the commands

ASSIGN STARTING-POINT

ASSIGN OPTIMISATION-BOUNDS

A further possibility is to re-formulate the event function so that it better assists the design point search. Often it helps to use a log-difference - log(resistance) - log(load) - if both resistance and load are always positive values.

2 The calculation of the multinormal probability fails.

In this case, the design point(s) have been found and the linearisation completed, but the resulting failure set is of a form so that the probability content of the set cannot be calculated. It might help in this case to change the convergence criterion to a smaller value, using the same command as above.

There is also the possibility that the event used in the analysis has probability zero or one because of a problem in the model. In these cases the model does not provide a limit state surface and therefore no design point.

4 EXECUTION OF PROBAN

Proban may be run in three different modes:

- In interactive line mode, using only character based input. The line mode facilities are described in Section 4.4.
- In interactive graphics mode with menus and dialog boxes, where input may be given using a mouse as well as the keyboard. The interactive graphics mode facilities are described in Section 4.5, but in addition this mode also gives access to the line mode facilities. It requires a work-station or an X-terminal running the OSF/MOTIF window system
- In batch mode, which uses the line mode syntax and facilities.

The start up of Proban in the three different modes is described in Section 4.1. This section also describes the files that Proban uses.

The program requirements and limitations are described in Section 4.2 and Section 4.3.

4.1 Program Environment

Proban is on Unix platforms delivered as an executable and an object file to be linked with user developed code. On NT platforms, the delivery is an executable and a DLL (Dynamic Link Library) for functions. The user replaces the DLL when he wants to run his own coded functions.

The Unix version requires that the Motif window manager is installed

Proban supports both graphics and line mode execution of the program.

How to start the program in the different modes is described below.

4.1.1 Command Line Arguments

It is possible to specify command line arguments when starting Proban. The command line arguments are simply added to the usual command starting the program:

prompt> proban /NOHEADER/STAT=OLD/INT=LINE/C-F=test_in.jnl/FORCED-EXIT

Please note that:

1 Command line arguments and values can be abbreviated, as described in Section 4.4.4. However, other input will be accepted, and used when possible.

2 Each argument name must begin with a slash (/), and each argument value must be prefixed by an equal sign (=). Spaces can be freely distributed around the equal sign and before each slash.

3 Texts with blank space and special characters (e.g. file names) can be protected in quotes. Please note that some operating systems change the case of the input text if it is not protected in quotes.

4 If at least one of /PREFIX, /NAME and /STATUS is specified, the prompt for database and journal file file name is disabled, and defaults are used for any unspecified values.

5 Proban will issue a message when an error is found in the command line specification.

Table 4.1 Command line arguments

/HEADER=SHORT	Give the usual start-up header (SHORT) or no start-up header (NONE).
/NOHEADER	Same as /HEADER=NONE.
/PREFIX=prefix	Specifies the database and journal file prefix.
/NAME=name	Specifies the database and journal file name.
/STATUS=status	Specifies the database and journal file status as OLD or NEW.
/INTERFACE=LINE	Start the program in line mode, ignoring the graphics user interface.
/INTERFACE=WINDOW	Start the program in graphics mode.
/COMMAND-FILE=filename	Read the specified command input file just after the database has been opened and initialised.
/NOCOMMAND-FILE	Do not read an initial command input file.
/FORCED-EXIT	Exit Proban after the database has been opened and initialised, and any initial command file has been read.
/NOFORCED-EXIT	Disable the forced exit.
/COMPANY-NAME=value	Specifies the header in the display (see also SET COMPANY-NAME).
/PRINT-FORMFEED=value	Use FORTRAN or ASCII formfeed character on LIS files.
/PLOT-COLOUR=value	Specifies the plot colour (see also SET PLOT COLOUR).

Table 4.1 Command line arguments

/PLOT-FORMAT=format	Specifies the plot file format (see also SET PLOT FORMAT).
/PLOT-PAGE-SIZE=value	Specifies the plot page size (see also SET PLOT PAGE-SIZE).
/DISPLAY-COLOUR=value	Specifies display colour (see also SET DISPLAY COLOUR).
/DISPLAY-DEVICE=device	Specifies display device (see also SET DISPLAY DEVICE).

4.1.2 Starting Proban in Graphics Mode

To start Proban in graphics mode, the computer must be running under the Motif window manager.

Proban reads a resource file with the name faceitClass (on Unix systems, note the use of upper- and lower-case letters). This file is placed in the directory where private X application resource files are kept, often the home directory.

Proban must use a fixed width font, otherwise the messages and prints will be misaligned.

If running on a Unix system, the command to be used to start Proban in graphics mode is simply:

```
prompt> proban
```

If running on an NT system, the command to be used to start Proban in graphics mode is simply:

```
prompt> proban or proban.exe
```

If running on an NT system, notice that the funclib.dll containing the functions must be in the user LIB path or on the same directory as the executable.

Proban responds by opening the main window, and overlaying it with a dialog box requesting the database file prefix, name and status, provided that none of these were specified as command line arguments (see Section 4.1.1).

Note that the default status is Old. Type in the file prefix and name, and select the proper status, then press the OK button (or type <Return>). Pressing the Cancel button will abort the session.

If the file specification is somehow in error, Proban will give an error message and keep the start-up dialog box open for a new file specification.

If the file specification is correct, Proban will open the database file (with extension ".MOD") and a journal file with the same prefix and name (but with extension ".jnl"). Proban can now be operated as described in Section 4.5 Using the Graphics Mode User Interface.

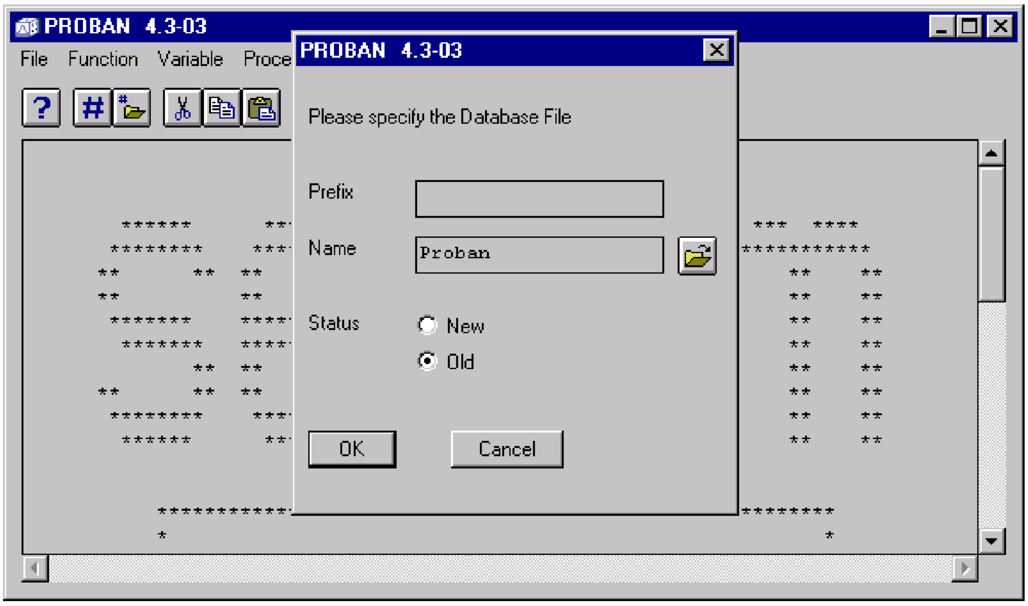


Figure 4.1 The program start-up dialog box

To exit the program, choose the Exit option under the File menu. Proban will then close all open files and terminate execution.

4.1.3 Starting Proban in Line Mode

A line mode session will not give access to the interactive graphics mode capabilities. The program runs in the terminal (window), and commands are typed on the input line.

To start Proban in line mode, specify /INTERFACE=LINE as a command line argument (see Section 4.1.1).

After a short while, a heading, similar to the one shown below, is echoed on the screen (provided that / NOHEADER was not specified on the command line).

```
 ******      ******       ******       ******      ** ***  ****
 ********    ********     ********     ********     *************
**      **  **      **    **      **    **      **     **    **    **
**          **      **    **                    **     **    **    **
 *******    **********    *******      *********    **    **    **
  *******   *********      *******   **********    **    **    **
        **  **                  **    **      **    **    **    **
**      **  **      **    **      **    **      **     **    **    **
 ********    ********     ********     *********    **    **    **
  ******     ******       ******       ******  **    **    **    **


   *************************************************************
```

```
*                                                          *
*                      P R O B A N                         *
*                                                          *
*              Probabilistic analysis system               *
*                                                          *
************************************************************

            Marketing and Support by DNV Sesam

  Program id   : 4.2-01                   Computer         : DEC 3000 Model 400
  Release date : 11-JAN-1996              Impl. update     :
  Access time  : 11-JAN-1996 15:06:55     Operating system : VMS V6.1
  User id      : OLES                     CPU id           : 0858461026
                                          Installation     : DNVS GRID

  Copyright DET NORSKE VERITAS SESAM AS, P.O.Box 300, N-1322 Hovik, Norway
```

Proban then invites the user to enter the model file name (more information in Section 4.1.2) through the following prompt;

```
Database file prefix ? / /
Database file name? /Proban/
```

No extension should be given since this file has a predetermined extension (For NT and UNIX installations this is .mod). The file name Proban (i.e. Proban.mod) is offered as a default.

```
Database File Status? /OLD/ NEW
```

If the Proban database file already exists the default OLD should be given. If the database is to be created, the answer is NEW. See also Section 4.1.2.

Note that if at least one of /PREFIX, /NAME or /STATUS is specified as a command line argument, the prompts for these values will be ignored, and the value(s) that are not specified will be given defaults.

This start-up has opened a new database file, called Proban.mod and a new journal file, called Proban.jnl. If the file specification is incorrect, Proban will reissue the prompt for the database file prefix.

Typing a double dot (..) during the start-up phase will abort the program.

The facilities that are available in line mode are described in Section 4.4.

To exit the program, type the EXIT command. This will close all files and exit the program.

4.1.4 Starting Proban in a Batch Run

Using command line arguments (see Section 4.1.1) is the simplest way to execute Proban in batch. If proban is the command that executes the program, the command to run test_in.jnl in batch could be:

proban /NAME=TEST/STAT=NEW/INT=LINE/C-F=test_in.jnl/FORCED-EXIT

Note that it is necessary to use the line mode interface, and that the forced exit tells the program to exit when the command input file has been read. This command assumes that the program is started at the directory for both the database file and the command input file.

The command can be enclosed in a batch command file (script). During a run, Proban reads commands from standard input, so the commands can be typed into the batch file after the program start-up.

On a UNIX platforms the user may create a batch input file, e.g. proban and then issue one of the commands below in order to execute PROBAN as a background process

```
prompt> proban < Proban.inp > Proban.log &
```

or:

```
prompt> proban /NAM=TEST/STA=N/INT=L/C-F=test_in.jnl/F-EX > Proban.log &
```

The header and messages given by Proban will appear on the log file.

On an NT platform, the background process requires that the script is coded in a proban.bat file.

4.1.5 Files and Data Safety

Proban makes use of the files shown in Table below.

File type	Extension	PROBAN		Format
		Reads from	Writes to	
DATABASE	.mod	YES	YES	Binary
JOURNAL	.jnl	NO	YES	ASCII
COM.INPUT	.jnl	YES	NO	ASCII
PRINT	.lis	NO	YES	ASCII
PLOT	varies	NO	YES	Binary/ASCII

The **DATABASE** (also called MODEL file) is a direct access file that is used to keep the probabilistic model and results. It has the extension: “.mod”.

The **JOURNAL** file is used to keep a log of most of the commands that are accepted during a Proban session. If an existing (OLD) database is opened, the journal will be appended to the corresponding old journal file if this exists. The journal file has the extension “.jnl”.

The **COMMAND INPUT** file is used to read commands and data into Proban. The usage of command input files is described in Table 4.4.2. The default extension of a command input file is “.jnl”, but this default is not used if another extension is specified.

The **PRINT** file is used to keep output from the PRINT command when the print destination is set to FILE. The extension of the print file is “.lis”. The print file name and settings is specified using the command: SET PRINT. It is possible to use more than one print file during the same Proban session, but only one can be open at a time.

The **PLOT** file is used to keep output from the PLOT command and from the DISPLAY command when the display destination is set to file. The plot file name and settings is specified using the command: SET PLOT.

The extension of the plot file depends on the plot format used. If the SESAM neutral format is used, the extension is “.plo”. Several other formats are available, including Postscript with extension “.PS”. It is possible to use more than one plot file during the same Proban session, but only one can be open at a time.

If the database file has been corrupted, the information may be reconstructed by use of the journal file. It is therefore recommended to take backup copies of the journal and database file at regular intervals.

4.2 Program Requirements

4.2.1 Execution Time

Most of Proban can be run interactively with no significant timing problems. However, the following situations may require so much computation time, that a batch run is advisable:

- Calculating a result with computational costly functions
- Calculating a result by use of extensive simulation

Because of an internal buffer limit, the database access performance may degrade considerably when a certain size of the database has been reached. It is not possible to predict exactly when this will happen.

4.2.2 Storage Space

The initial size of the program on NT is about. 4Mb

The initial size of the program on Unix is about. 9Mb

The initial size of the database is about. 230Kb

4.3 Program Limitations

The following limitations apply. See also the status list for current updates to this.

The names functions, variables and events are limited to 12 characters. All names are case insensitive when matched with input text.

Descriptive texts are in most cases limited to 50 characters.

There is a limit on the number of random variables that can be presented through the user interface (this limit does not apply to the number that can be stored in the database). The effect is to prohibit use of some commands and generate some error messages when the commands are used. The commands, that in particular may create problems, include

```
DEFINE PARAMETER-STUDY
ASSIGN SENSITIVITY
PRINT VARIABLE
RUN CROSSING-RATE-ANALYSIS
RUN FIRST-PASSAGE-PROBABILITY-ANALYSIS
RUN PROBABILITY-ANALYSIS
```

```
RUN DISTRIBUTION-ANALYSIS
RUN DETERMINISTIC-ANALYSIS.
```

4.4 Using the Line Mode User Interface

The line mode environment in Proban is very powerful. It has many features and provides a great flexibility to the user. This section describes the facilities one by one. Even when running graphics mode, the line mode environment is available through the command input line.

There are two modes of operation inside the line mode environment, called “command mode” and “programming mode”.

Command mode is the commonly used mode, it is used to give commands to Proban. A new input line always starts in command mode. To switch to/from programming mode inside an input line, type the dollar sign: $.

Programming mode is used basically to calculate numerical values. These values can then be used in a command if desired, or they can be viewed as results.

When moving through the commands, Proban will present a prompt, possibly followed by a default inside / /. The main command level is signified by the prompt: #. No default is presented here. The main commands are ASSIGN, CREATE etc. These are described in Chapter 5. When moving inside a command the prompt will change and a default may be presented.

Different items on the command line are separated by blank spaces, except if it is text that is protected inside quotes. In special cases, the blank space may be left out. Such cases are documented in the sections below.

Proban does not require line breaks anywhere. Thus several commands can be typed into the same command input line.

In the following, input typed by the user is shown in bold face while prompts given by Proban are shown as ordinary text.

4.4.1 How to get Help

Context sensitive help is available in command mode at any time using any of these methods:

Table 4.2 How to get help in line mode

Type: ?	to get a brief description of what Proban is expecting right now.
Type: <text>?	during a selection between alternatives to see all the alternatives that match <text>. <text> may contain wildcards or be an abbreviation.
Type: ??	to get a more descriptive help text, showing how to proceed.

There is also a HELP menu under the main menu, giving on-line access to the items that are described here.

4.4.2 Command Input Files

Line mode commands may be read from a file as well as typed directly into Proban. Such a file may contain any syntax that is allowed in line mode, including reading another command input file.

To read in a command input file, type an @ followed by the file name. To read parts of the file, specify the number of lines to read after the file name. If the file name does not have a suffix (i.e. a dot and the following part), Proban adds ".jnl" to the name.

Proban may have more than one command input file open at one time (i.e. you may reference a command input file from within another command input file). It will always read each file sequential, finishing the last opened file first. To get a list of the currently open files, type: @?

The last opened command input file may be closed explicitly by typing the @ followed by two dots: @..

When a command input file is being read, the lines read are echoed on the screen and logged on the journal file. Programming expressions are logged as comments and the resulting values are logged as part of the command. The @ command itself is not logged on the journal file.

If an error is found in a command input file, Proban stops reading the file and skips the remaining part of the line where the error was found.

Proban will also stop reading of a command input file if it finds a line containing only an @

The commands used to manipulate command input files are summarised below.

Table 4.3 Manipulation of command input files

@filename	Read the named file from the top. Reading will stop is an error if found, or at the end of the file, or if a line with only an @ is found. There may be one or more blank spaces between @ and the file name.
@filename <n>	Read <n> lines of the named file from the top. Reading will stop if an error is found, or if a line with only an @ is found. There may be one or more blank spaces between @ and the file name.
@	Continue reading the presently open file. Reading will stop if an error is found, or at the end of the file, or if a line with only an @ is found.
@ <n>	Continue reading the presently open file. Reading will stop if an error is found, or if a line with only an @ is found.
@..	Close the last opened command input file. There cannot be any blank space between @ and the dots.
@?	Show the name and status of the currently open command input file(s).

4.4.3 Accessing Default Values

Proban will in many cases supply a default value when input is requested. The default will be presented in / /. An example:

```
DEFINE ANALYSIS-OPTION PARAMETER-STUDY
Run Parameter Study Analysis? /ON/
```

The default may be accepted using one of the following methods:

Table 4.4 Input of default value(s)

<Return>	(i.e. an empty input line) to accept the current default.
: (colon)	to accept the current default. The colon must be preceded by a blank if it is not the first item on the command line. However, several colons may follow each other without intervening spaces.
; (semicolon)	to keep accepting defaults as long as they are presented, or until the command is complete. The semicolon must be preceded by a blank space if it is not the first item on the command line. However, several semicolons may follow each other without intervening spaces.

Please note that an empty line in a command input file will not be interpreted as a default. The colon and semicolon may be written into a command input file.

A colon or semicolon is never logged on the journal file. Instead, the substituted default values are logged.

4.4.4 Abbreviation and Wildcards

Proban offers two methods to short-cut selection of elements in a list: Abbreviation and the use of wildcards.

Alternatives up to hyphens can be abbreviated, as long as the abbreviation is unique. Thus, SUB-LEVEL-INTEGRATION may be abbreviated to any of: SU, S-L-I, S-LEV as long as the abbreviation is unique among the alternatives presented.

Wildcards consist of the following two characters:

Table 4.5 Wildcard characters

*	substitutes for any number of characters (including no characters).
&	substitutes for any one character. It must match exactly one character.

As an example, *y&&& matches xabycc1 and xy111 but not xaby11.

Abbreviation and wildcards may not be mixed in the same matching expression.

4.4.5 Input of a Text or a Name or a Numerical Value

Numerical values can be input in free format in Proban. Floating point numbers as: 1000, 1., .54, 1e-44, and .1e5 are all accepted.

Whole numbers can be specified as floating point numbers. Examples of whole numbers: 1000, 1., .1e4

Names may contain any alphanumeric character as well as the underscore (_) and the hyphen (-). An integer will be accepted as a name, but will not work when referenced in situations which permits numerical input. Good practice is to start a name with an alphabetic character. The maximal length of a name is documented with the command where the named object is created.

Text must be encapsulated in single quotes if it contains blank space(s) and/or special characters:

'This is a text containing 10 spaces and a single @'

4.4.6 Selecting a Single Alternative from a List

In many cases, Proban will require a selection of a single alternative from a list. An example is right at the start, at the main prompt: #, where the main commands are presented for selection. The selection need not be a selection between commands, it could also be a selection between named objects or between numerical values.

In selection of a single value, abbreviation is allowed, but wildcards cannot be used. An exact match is always preferred. Thus it is possible to select an item that is an abbreviation of another item in the list by typing the item exactly.

A single question mark: ? will show all items in the list. Prefixing the question mark with a text: <text>? will show all items in the list matching <text>.

The input text may be typed in upper case or lower case. Proban disregards the case of the text when comparison is made.

The input text used to make the selection is not logged on the journal file. Instead, the selected value is logged as it is presented in the list.

4.4.7 Selecting Several Alternatives from a List

In some cases, a list of items is presented, from which one or more items can be selected. An example is the DISPLAY DISTRIBUTION command, where a number of names may be selected for display.

In this selection, both wildcards and abbreviation may be used (but not inside the same text).

The syntax for the selection allows for more flexibility than in the single selection case, because it may be of interest to keep modifying the selection for some time before accepting it. The selection process consists of one or more selection operations, each of which follow the syntax described below. If more than one operation is required to complete the selection, the selection must be enclosed in parentheses: ( )

The syntax for a single selection operation is:

Table 4.6 Selection of several alternatives from a list

INCLUDE <text>	Include the item(s) matching <text> in the selection. Set the default status to INCLUDE. Any items specified after this will be included in the selection until the status is changed.
ONLY <text>	Set the current selection to the item(s) matching <text> Set the default status to INCLUDE. Any items specified after this will be included in the selection until the status is changed.
EXCLUDE <text>	Exclude the item(s) matching <text> from the selection. Set the default status to EXCLUDE. Any items specified after this will be excluded from the selection until the status is changed.
<text>	Include or exclude the items matching <text>, depending on the default status. The initial default status is INCLUDE.
GROUP <from> <to> <step>	In the case of a selection of numerical values, or of a selection between names (which can be integer values), the <text> can be substituted with this interval expression which expands to the values: <from>, <from> + <step>, <from> + 2*<step>, ...up to but not exceeding <to>.

When a default selection is being presented, or if the left parentheses has been typed as input, Proban presents the right parenthesis as default: /)/.

A single question mark: ? will show all items in the list, listing the currently selected items in parenthesis. Prefixing the question mark with a text: **<text>?** will show all items in the list matching <text>.

Example 4.1

```
DISPLAY DISTRIBUTION *
```

will display all distributions currently stored in the database.

```
DISPLAY DISTRIBUTION ( * EXCLUDE B* )
```

will display all distributions except those with names starting with B.

4.4.8 Entering a Vector or a Matrix of Values

The syntax for entering a vector or a matrix of values is an extension of the syntax for selecting values from a list. In this case there is no fixed list to select from. Instead the items are inserted and manipulated as the vector/matrix is entered.

The term vector is used for the case where the input is one dimensional. The term matrix is used for the case where the input is multidimensional. Like a vector is built up from single items, a matrix is built from rows. There cannot be an unequal number of items in two different columns of a matrix.

The input of a vector/matrix is consists of one or more operations. If more than one operation is required (as it most likely will be), they must be enclosed in parentheses.

The syntax of one operation is (<row> refers to a single value in a vector or to a row in a matrix):

Table 4.7 Entering a vector of matrix of values

INCLUDE <row>	Include the specified <row> as the last row. Set the default status to INCLUDE. Until the status is changed, rows that are entered will be added at the end.
EXCLUDE <row>	Exclude the specified <row>. Set the default status to EXCLUDE. The next row(s) that are entered will also be excluded until the default status is changed. Wildcards may be used to specify <row>. All matching rows will be excluded.
ONLY <row>	Include only <row> in the matrix, clearing any previous contents first. Set the default status to INCLUDE.Until the status is changed, rows that are entered will be added at the end.
INSERT-BEFORE <row1> <row2>	Insert <row2> before <row1>. Set the default status to INSERT-BEFORE. Until the status is changed, rows will be keep being inserted before <row1> (immediately after the last row entered). Wildcards may be used to specify <row1>, provided that one row is matched uniquely.
OVERWRITE <row1> <row2>	Overwrite <row1> with <row2>. Set the default status to OVERWRITE. The next row(s) that are entered will continue overwriting until the default status is changed, scrolling down as they do so. When the last row has been overwritten, the default status is changed to INCLUDE. Wildcards may be used to specify <row1>, provided that one row is matched uniquely.
LIST	List the contents of the matrix.
<row>	Insert, Exclude or overwrite, using <row>, depending on the default status. The initial default status is INCLUDE.

When a default vector/matrix is being presented, or if the left parenthesis has been typed as input, Proban presents the right parenthesis as default: /)/.

A single question mark will show the possible alternatives in the matrix.

Use LIST to see the rows in the matrix.

4.4.9 Setting and Clearing Loops in a Command

When a command is completed, then Proban by default goes back to the main command level: #. If a command is to be repeated many times in slightly different versions, it may be desirable to go back to an intermediate command level rather than to the main command level. This is accomplished by typing LOOP when the intermediate command level to be repeated from is entered. The loop is ended by typing END at the command level repeated from, or by aborting the command by using the double dot (..) .

Example 4.2

```
DEFINE CONTINUOUS-PROCESS
LOOP
  ANALYSIS-OPTION etc
```

```
  DURATION etc
  STARTING-TIME etc
END
```

4.4.10 Inserting a Command into Another Command

It is possible to insert a command at any point while in command mode (not in programming mode). This is done by simply typing the main prompt: # followed by the inserted command.

Proban will finish the new command, and then return to the command level in the previous command, where the new command was inserted.

This is useful e.g. for catching up on settings or definitions that was forgotten while inside a PRINT or DISPLAY command, or for printing out objects to see what they contain. The following examples illustrate this:

```
DISPLAY FUNCTION DIFFERENCE # PRINT FUNCTION DESCRIPTION DIFFERENCE
        etc
```

The same command cannot be entered recursive, e.g. a DISPLAY FUNCTION command cannot be issued inside another DISPLAY FUNCTION command.

Commands can be nested this way to as many levels as desired. However, to nest with more than one level may be confusing and is not recommended. The current status may be seen by typing: -?.

4.4.11 Aborting All or Parts of a Command

To abort a command, type two dots after each other: .. . Please note that all entries on the command line up to the double dot will be processed before the command is aborted.

The double dot clears all loops and previous input in the command and then presents the main prompt: # .

A double dot is only logged if a part of the current command has already been written to the journal file.

To abort parts of a command, going back to the last LOOP or to the point of a left parenthesis in a multiple selection or a vector or a matrix, type: <<< .

CtrlC may also be used to abort a command (hold the Control key while typing C). Usage of CtrlC will throw away all of the input of the command line as well as abort the command. Unlike the double dot, the input before the CtrlC is not processed. CtrlC may also be used to abort a running analysis.

4.4.12 Access to the Operating System

It is possible to issue a command to the operating system at any level in a Proban command (not from programming mode). This is done by typing an exclamation mark: ! followed by the operating system command. Everything on the input line after the exclamation mark is sent to the operating system.

The following example, taking from a run on a Unix computer, will list all journal files on current directory.

```
!ls *.JNL
```

The command below spawns a sub process on a Unix system and must be terminated by use of the command: exit.

```
!sh
```

The facility is very useful for obtaining directory listings, editing files (e.g. input files), spawning into the operating system to do more complicated tasks, etc.

The facility is also available from the command input line in graphics mode, but, when used here the output from the operating system will appear in the terminal window from which Proban was started.

4.4.13 Appending Input Lines

After receiving an input line, Proban processes the input, unless told otherwise. The way to suspend processing of an input line is to type a backslash: \ as the last character in the line. Proban then issues the append prompt: >>.

4.4.14 Viewing the Current Status of a Command

Some commands are long, and it may be difficult to keep track of what has actually been given as input. In other cases where commands have been inserted, it may be useful to see what the current command(s) actually look like to Proban. This is achieved by use of the command: -?.

4.4.15 Comments

A comment may be typed anywhere in a command while in command mode (not in programming mode). Comments are prefixed by the percent sign: %. Everything from the percent sign to the end of the line is treated as a comment. A comment need not be the first item on a line.

Example 4.3

```
CREATE VARIABLE Time ' ' TIME%In seconds
% This is a comment.
```

4.5 Using the Graphics Mode User Interface

The Proban graphics environment offers a main window with the following parts (from top to bottom):

- Title bar. This is the name of the program that is being run.
- Main menu. This menu gives access to all the commands of Proban.
- Short-cut buttons. The first three toggles command input mode on and off, reads a command input file and closes a command input file. This last button is only active when a command input file is open. The last three buttons will cut, copy and paste texts to and from the text input areas of Proban.
- Message area. This is used to show messages to the user, plus commands that have been typed into the command input line, as well as those that have been read from command input files.
- Command input line This line contains the prompt for line mode input (showing the default when this is available), followed by a field which is used to type line mode commands. All facilities that are described in Section 4.4 are available through this line.

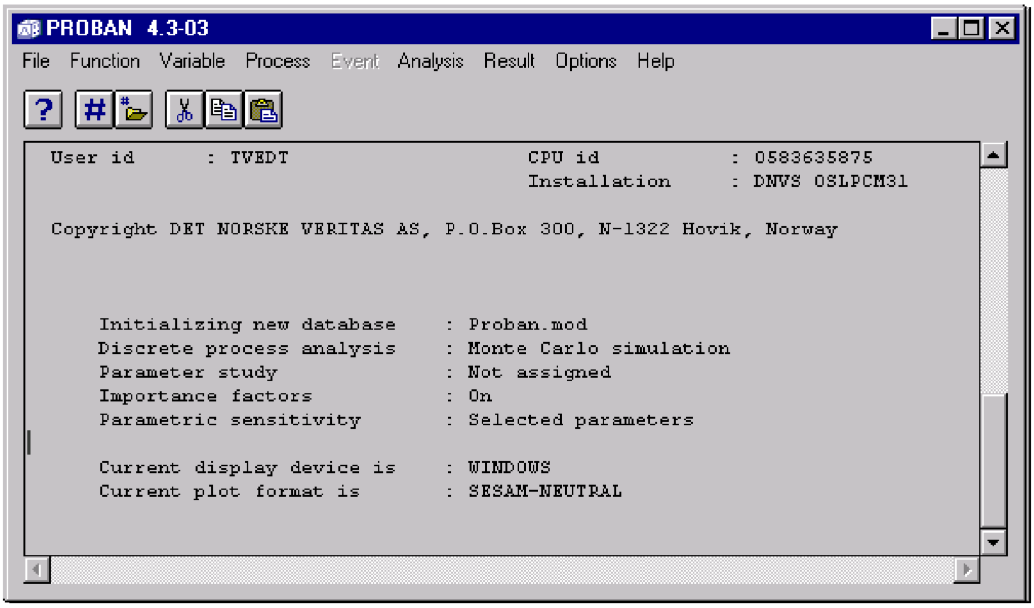


Figure 4.2 The main dialog window at start-up

In addition to the parts seen in Figure 4.2, the graphics area and command line area may be visible, as shown in Figure 4.3.

- The command line and prompt at the bottom, as well as the command list at the right and the six short-cut buttons are used to give line mode commands to Proban. A command can be entered by clicking in the command list or by typing text in the command line, followed by <Enter>. The short-cut buttons all have explanatory text attached, visible when the mouse pointer is paused over the button. Two extra buttons appear when a command input file is open.

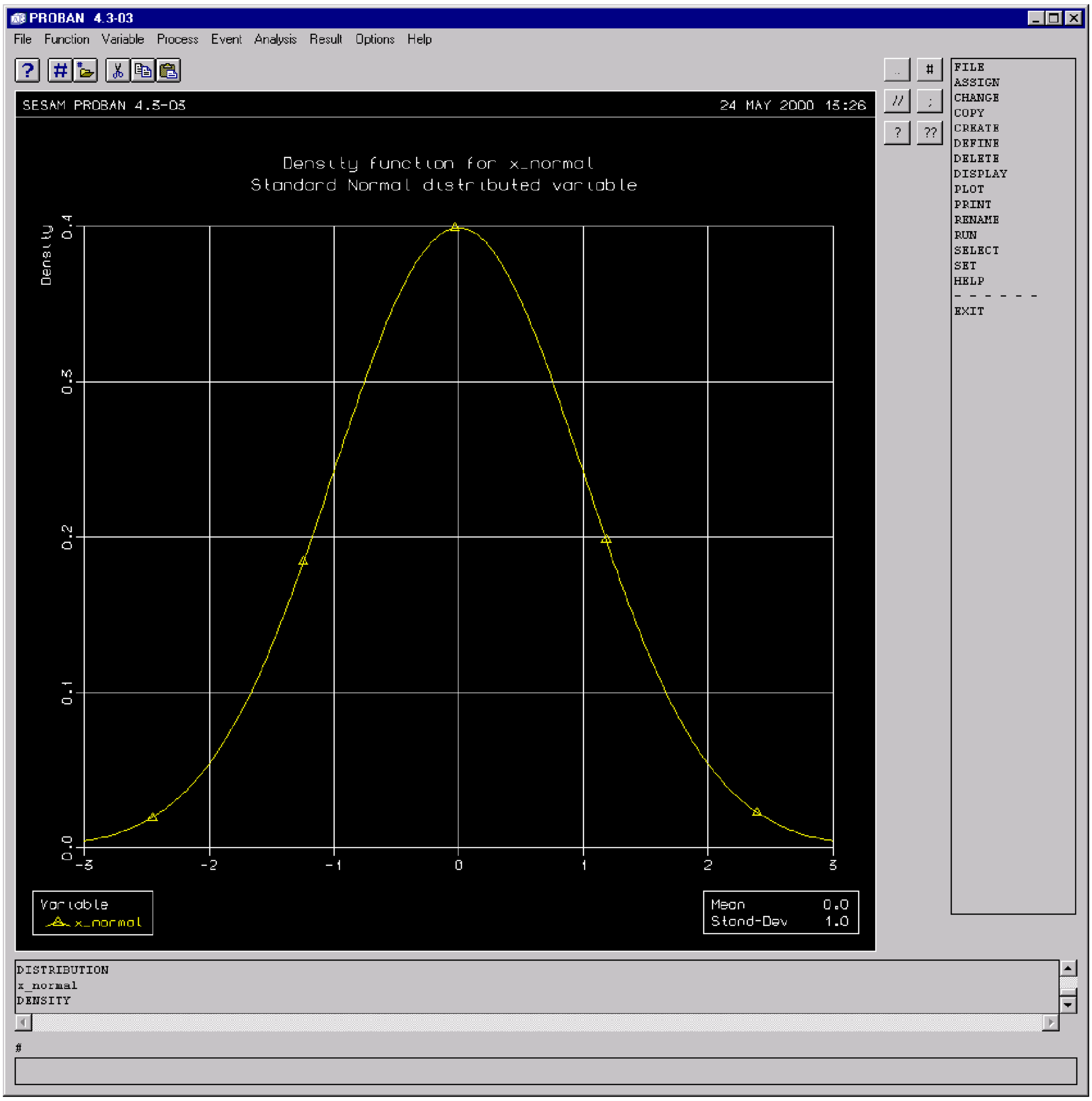


Figure 4.3 The main window with graphics area and line mode command input area

If the main window is iconised, all the open dialog boxes disappear into the icon. They pop up again when the main window is popped up. In addition to this, the graphics environment consists of:

- Pulldown menus. These are pulled down from the items in the main menu. They are activated by clicking on an item in the main menu with the left mouse button, or by holding the left mouse button down on an item in the main menu. Similarly, some of the items in a pulldown menu may have a sub menu sliding sidewards from the parent menu. To select an item in a pulldown menu, click on it or drag the mouse pointer to the item and release the button.

- Dialog boxes. Much of the user interaction will happen through dialog boxes. Those items in the pull-down menus that have three dots following the item label, all open a dialog box when selected. The dialog box is described more fully in Section 4.5.3.

- Print window. After the first Print command has been issued, a print window will pop up. This is a scrollable window, that contains all the output from the Print command, that is directed to the screen. The window has a limited buffer, so if a single print command generates excessive amounts of print, some of it may disappear out of the top of the window. The print window may be iconised separately from the main window. It is possible to print inside an iconised print window. It does however not pop up automatically from an iconised state when something is printed.

4.5.1 How to Get Help

There is a Help menu under the main menu, which contains useful on-line information.

Context sensitive help is available through a Help button (the F1 button on some computers). When an entry in a dialog box (e.g. a text input field or a scrtollable list) is active, pressing the Help button will often display a context sensitive help text in a separate window.

4.5.2 Tear-Off Menus

When using Motif version 1.2 or higher, a pulldown menu can be torn off and displayed in a separate window. This is very useful for accessing commonly used dialog boxes. The menu is torn off by clicking on the stipulated line at the top of the menu (if no such line is visible, the menu cannot be torn off). To close the menu, select the “Close” entry in the menu at the upper left corner of the window frame.

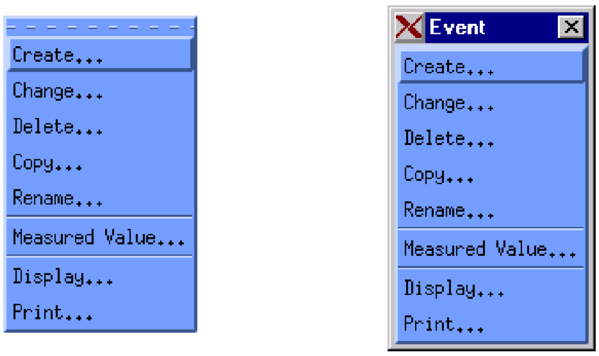


Figure 4.4 Tear-off pulldown menu before and after it is torn off

4.5.3 Dialog Boxes and their Contents

A dialog box is used to pass information from the user to Proban. Most dialog boxes also present the current defaults, and thus may be used to pass information from Proban to the user.

The typical entries in a dialog box are: **Input fields, Menus** and **Pushbuttons**.

An **Input field** can contain a text, a name or a numerical value. The Set Plot dialog box contains two input fields: the file prefix and the file name description.To type into the field, place the pointer in the field and press down the left mouse button. In some input fields, the text can be longer than the width of the field as shown in the dialog box. The text will then scroll if typed beyond the width of the input field.

Menus come in four different types: **Togglebuttons, Radio boxes, Option menus** and **Scrollable lists**. Selecting in a menu may cause considerable changes in the layout of the dialog box. This will depend on the dialog box in use.

A **Togglebutton** is a button that has two states: On and Off. One example is given in the Set Plot box, where the Colour button is Off. To switch the status of the button, place the pointer on the button and press down the left mouse button.

A **Radio box** is a collection of togglebuttons, where only one button can be active. All buttons are visible on the screen simultaneously. An example is the Type buttons in the Display Distribution dialog box. To select a button, place the pointer on the button or on its corresponding label and press down the left mouse button.

An **Option menu** is similar to a radio box, in that it presents a number of alternatives, of which only one can be active. It is however operated differently. To display the menu, place the pointer on the button showing the active alternative and press down the left mouse button. To select an alternative from the menu, place the pointer on the alternative and press down the left mouse button. Alternatively, display the menu but keep the mouse button down. Then move the pointer through the menu to the selected alternative and then release the mouse button. The Type menu of the Variable dialog box is an example of an option menu.

A **Scrollable list** is a list of alternatives, that is presented in a scrollable box. Such a menu is used in order to save space, or because the items in the list cannot be predicted before the menu used. A scrollable list is either a single selection list, or a multiple selection list. Use the scrollbar to manoeuvre through the list. In a single selection list, place the pointer on the desired alternative and press down the left mouse button. In a multiple selection list, place the pointer on the first desired alternative and press down the left mouse button and keep it down. Then drag the pointer through the list and release the button when the selection is ready. To modify an existing selection in a multiple selection list, hold the Control key down and make a selection as described above. The alternatives selected this way then reverse their selection status. Selected values are marked by highlighting. The Distribution list in the Create Variable dialog box is an example of a single selection scrollable list. The Function list in the Print Function Description dialog box is an example of a multiple selection scrollable list.

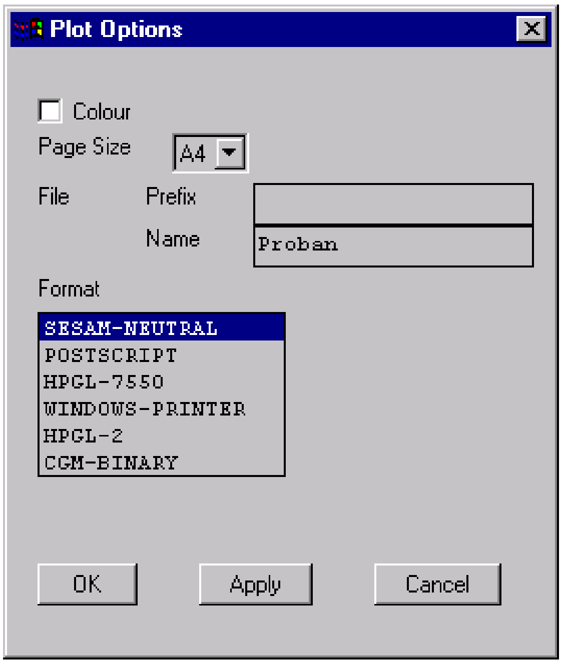


Figure 4.5 The Set Plot dialog box

A **Pushbutton** is a button, that causes an action when it is clicked on.

OK, Apply and Cancel buttons are represented in the Set Plot box shown above. All dialog boxes have a standard set of buttons at the bottom of the box. These buttons are described later in this section.

If the label of a pushbutton is followed by three dots, the button will open a new dialog box. The Assign dialog boxes often contain pushbuttons that provide a short-cut to boxes placed under the Select main command.

In addition to these items, there are a few more complex input items, that are described in the following sections.

4.5.4 The Standard Buttons in a Dialog Box

A dialog box will contain one or more of these standard buttons, placed at the bottom of the box:

Table 4.8 The standard buttons of a dialog box

OK	Accept the contents of the box and close the box. The box will not be closed if the processing of the contents of the box gives an error.
Apply	Accept the contents of the box. The box is not closed.
Cancel	Close the box without accepting the contents.
Close	Close the box without accepting the contents.

Table 4.8 The standard buttons of a dialog box

Update	Update the contents of the box to correctly represent information changed elsewhere.
Help	Provide context sensitive help

Most dialog boxes have a default pushbutton, that is activated by typing <Return> when the dialog box is active. This pushbutton is usually the OK or the Apply button. The default button will be highlighted or framed.

4.5.5 Entering a Prefixed List

The prefixed list is used to enter a number of values, that is unknown until the time the box is used, where each value has a prefix (or prompt). It is, for example, used to input distribution parameters, function arguments and starting point values.

In line mode, the list is simply traversed sequential from top to bottom. In graphics mode, the accompanying input field (located just below the box) is used to input and change values. The procedure used to change or input a value is:

- Select the corresponding row in the box. Doubleclick on the row if desired to transfer the current value to the input field. If no row is selected, the first row is implicitly used.
- Type the new value in the input field.
- Hit <Return> in the input field to transfer the value to the box. The next row in the box will then be selected and the input field will be cleared.

Thus it is possible to input values sequential into the box by clicking on the input field and then typing the values one by one, with each value followed by a <Return>.

4.5.6 Entering a Vector or a Matrix of Values

In many cases a vector or matrix of values must be input. An example is entering fractiles, cumulative probabilities and weights in the CREATE VARIABLE ... FITTED-DISTRIBUTION command.

The graphics mode input of this is quite flexible. The values are presented in columns in a scrollable box. Under the box is one input field for each column in the matrix (one field if it is a vector). Under the input field(s) are two rows of buttons, that are used to manipulate the contents of the box.

Type values into the input fields, and hit <Return> in the last (bottom) field. The values are then inserted at the bottom, or before the selected row, or overwrites the selected row, depending on the default status. The

initial status is Include, which inserts values at the bottom. The input fields are cleared after the insertion is complete. Instead of pressing <Return>, a button may be pressed. The effect of this is:

Table 4.9 Entering a vector or matrix of values in graphics mode

Include	Include the values in the input field(s) at the bottom, then clear the input fields. Sets the default status to Include.
Exclude	Exclude all selected rows from the matrix/vector. Sets the default status to Exclude.
Overwrite	Overwrite the selected row with the contents of the input fields. Only one row can be selected in the scrollable box. The next row (if any) will then be selected, and the default status will be set to Overwrite. The input fields will be cleared.
Insert before	Insert the contents of the input fields before the selected row. Only one row can be selected in the scrollable box. The default status will be set to “Insert before”. The input fields will be cleared.
Clear	Clear the contents of the matrix. NOTE: There is no way to get the cleared contents back, other than perhaps cancelling/closing the dialog box and opening it again.
Help	Pressing this is equivalent to pressing the help button while the scrollable box has the input focus. It provide on-line access to a description of how to use the matrix/vector.

4.5.7 Journalling from Graphics Mode

All commands that are accepted from graphics mode are logged on the journal file. The commands are logged in a format that can be read into the corresponding line mode command.

There is one case, that deserves attention:

Some dialog boxes contain many line mode commands. An example is the Set Plot dialog box (Figure 4.4 and Figure 4.5). Since all the visible contents of a dialog box are selected when the OK or Apply button is pressed, even if only parts of the box has been changed, all possible commands in the box will be logged.

Pressing the OK or Apply button in this box will generate the following log:

```
SET PLOT COLOUR OFF
SET PLOT FILE ' ' PROBAN
SET PLOT FORMAT SESAM-NEUTRAL
SET PLOT PAGE-SIZE A4
```

5 COMMAND DESCRIPTION

This chapter describes all the commands available in Proban.

As described in Chapter 4, Proban has two user interfaces: A graphical user interface (also called graphics mode) and a text based command interface (also called line mode).

The first section of this chapter lists the correspondence between the pulldown menus available in the graphical user interface and the line mode commands.

The line mode input is journalled, also when the graphical user interface is used. The line mode input is therefore described in full in this chapter. The second section lists the line mode commands alphabetically.

The hierarchical structure of the line-mode commands and numerical data is documented in this chapter by use of tables. How to interpret these tables is explained below. Examples are used to illustrate how the command structure may diverge into multiple choices and converge to a single choice.

In the example below command A is followed by either of the commands B and C. Thereafter command D is given. Legal alternatives are, therefore, A B D and A C D.

<table>
<tr><td rowspan="2">A</td><td>B</td><td rowspan="2">D</td></tr>
<tr><td>C</td></tr>
</table>

In the example below command A is followed by three selections of either of commands B and C as indicated by *3. For example: A B B B, or: A B B C, or A C B C, etc.

<table>
<tr><td rowspan="2">A</td><td>B</td><td rowspan="2">*3</td></tr>
<tr><td>C</td></tr>
</table>

In the example below the three dots in the left-most column indicate that the command sequence is a continuation of a preceding command sequence. The single asterisk indicate that B and C may be given any number of times. Conclude this sequence by the command END. The three dots in the right-most column indicate that the command sequence is to be continued by another command sequence.

<table>
<tr><td rowspan="3">...</td><td rowspan="3">A</td><td>B</td><td rowspan="2">*</td><td rowspan="3">...</td></tr>
<tr><td>C</td></tr>
<tr><td colspan="2">END</td></tr>
</table>

In the example below command A is followed by any number of repetitions of either of the sequences B D and C D. Note that a pair of braces ({ }) is used here merely to define a sequence that may be repeated. The braces are not commands themselves.

<table>
<tr><td rowspan="2">A</td><td rowspan="2">{</td><td>B</td><td rowspan="2">D</td><td rowspan="2">}*</td></tr>
<tr><td>C</td></tr>
</table>

The characters A, B, C and D in the examples above represent parameters being line-mode COMMANDS (written in upper case) and numbers (written in lower case). All numbers may be entered as real or integer values. Brackets ([ ]) are used to enclose optional parameters.

A parameter followed by a '+' signifies a selection of one or more numerical values, names or texts from a list of items.

A parameter followed by a '*' signifies one or more alphanumeric or numerical values of the same type. These values are entered as a prefixed list.

Note: Line mode commands are in this chapter presented in upper case including hyphens. In graphics mode the commands appear in mixed case and without hyphens.

Note: Graphics mode commands that are irrelevant at a given time are masked out (shown grey in graphics mode).

Use of Proban in graphics mode is described in Section 4.5. Tutorial examples of line mode command input are given in Chapter 3.

The HELP command is not described here. It is intended purely to serve as on-line help. Usage of the HELP command is not logged. When in doubt how to do things try the HELP command.

5.1 Graphical User Interface Menus

The pulldown menus of the graphical user interface are listed here from left to right and top to bottom, together with the line mode commands to which they correspond. The line mode commands can be found alphabetically in the next section.

Please note that some line mode commands are available through more than one pulldown menu. This is purely for convenience, and does not affect the journalling of these actions. Some dialog boxes are also available through short-cut buttons inside other dialog boxes.

5.1.1 The File Menu

This pulldown menu contains file manipulation commands and the command used to exit Proban.

Open	FILE OPEN
Plot	PLOT
Exit	EXIT

5.1.2 The Function Menu

Create Function	CREATE FUNCTION
Change Function	CHANGE FUNCTION
Delete Function	DELETE FUNCTION
Copy Function	COPY FUNCTION
Function Option	ASSIGN FUNCTION-OPTION
Select Library	SELECT FUNCTION-LIBRARY
Display Function	DISPLAY FUNCTION
Presentation Options	DEFINE PRESENTATION FUNCTION
Print Description	PRINT FUNCTION DESCRIPTION
Print Formula	PRINT FUNCTION FORMULA
Print Response Surface	PRINT FUNCTION RESPONSESURFACE
Print Value	PRINT FUNCTION VALUE
Print Gradient	PRINT FUNCTION GRADIENT
Print Library	PRINT FUNCTION LIBRARY

5.1.3 The Variable Menu

This menu contains commands used to define random variables.

Create Variable	CREATE VARIABLE
Change Variable	CHANGE VARIABLE
Delete Variable	DELETE VARIABLE
Copy Variable	COPY VARIABLE
Rename Variable	RENAME VARIABLE
Extreme Type	ASSIGN EXTREME-VALUE
Function Option	ASSIGN FUNCTION-OPTION
Conditioning	ASSIGN CONDITIONING
Display ->	
One Dimensional Distribution	DISPLAY DISTRIBUTION
Fitted Distribution	DISPLAY FITTED-DISTRIBUTION

Print ->	
Print Basic Information	PRINT VARIABLE
Print Distribution	PRINT DISTRIBUTION
Print Correlation	PRINT CORRELATION
Correlation ->	
Correlate Variables	ASSIGN CORRELATION

5.1.4 The Process Menu

Continuous Process->	
Time Derivative	ASSIGN CONTINUOUS-PROCESS TIME-DERIVATIVES
Stationary Process Duration	DEFINE CONTINUOUS-PROCESS DURATION
General Process Start Time	ASSIGN CONTINUOUS-PROCESS STARTING-TIME
General Process Duration	ASSIGN CONTINUOUS-PROCESS DURATION

5.1.5 The Event Menu

This menu contains commands used to model events.

Create Event	CREATE EVENT
Change Event	CHANGE EVENT
Delete Event	DELETE EVENT
Copy Event	COPY EVENT
Rename Event	RENAME EVENT
Measured Value	ASSIGN MEASURED-VALUE
Display Event	DISPLAY EVENT
Print Event	PRINT EVENT

5.1.6 The Analysis Menu

This menu contains commands used to set up and execute probabilistic and deterministic analyses in general. Results from such an analysis are examined by use of the “Result” menu.

Select Analysis Method	SELECT ANALYSIS-METHOD
General Analysis Setup	DEFINE ANALYSIS-OPTION

Sensitivity Calculation ->	
Selection	ASSIGN SENSITIVITY VARIABLE
Increment	ASSIGN SENSITIVITY INCREMENT
Parameter Study	DEFINE PARAMETER-STUDY
Run Analysis ->	
Probability	RUN PROBABILITY-ANALYSIS
Distribution	RUN DISTRIBUTION-ANALYSIS
Deterministic	RUN DETERMINISTIC-ANALYSIS
Continuous Process	RUN CONTINUOUS-PROCESS
Restart Simulation	RUN RESTART
FORM/SORM Analysis Setup ->	
General FORM/SORM Setup	DEFINE FORM-SORM
Optimization Bounds	ASSIGN OPTIMISATION-BOUNDS
Starting Point	ASSIGN STARTING-POINT
Nested Analysis, Optimization	DEFINE FORM-SORM NESTED-ANALYSIS
Nested Analysis, General	DEFINE ANALYSIS-OPTION NESTED-ANALYSIS
Generated Distribution	DEFINE FORM-SORM GENERATED-DISTRIBUTION
Probability Simulation Setup	
Axis Orthogonal Simulation	DEFINE PROBABILITY-SIMULATIONAXIS-ORTHOGONAL
Directional Simulation	DEFINE PROBABILITY-SIMULATION DIRECTIONAL
Monte Carlo Simulation	DEFINE PROBABILITY-SIMULATION MONTE-CARLO
Distribution Analysis Setup	
Simulation	DEFINE DISTRIBUTION-SIMULATION
Mean Value FORM	DEFINE MEAN-VALUE-FORM
Continuous Process Setup	
General Analysis Setup	DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTIONS

Sub Level Integration	ASSIGN SUB-LEVEL-INTEGRATION

Check Analysis Input

Probability	RUN INPUT-CHECK PROBABILITY-ANALYSIS
Distribution	RUN INPUT-CHECK DISTRIBUTION-ANALYSIS
Continuous Process	RUN INPUT-CHECK CONTINUOUS-PROCESS-ANALYSIS

Print Analysis Setup

Analysis Setup	PRINT ANALYSIS-SETTINGS
Parameter Study	PRINT PARAMETER-STUDY
FORM/SORM Starting Point	PRINT STARTING-POINT

5.1.7 The Result Menu

This menu contains commands used to access results created while running probabilistic or deterministic analysis.

The results created during general probabilistic or deterministic analysis must be accessible through this menu.

Save Result	SAVE RESULT
Select Result	SELECT RESULT
Delete Result	DELETE RESULT
Rename Result	RENAME RESULT
Result Presentation	DEFINE PRESENTATION RESULT

Display Result ->

Distribution	DISPLAY RESULT DISTRIBUTION
Importance Factors	DISPLAY RESULT IMPORTANCE-FACTORS
Parameter Study, Main Result	DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT
Parameter Study, Importance	DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR

Print Result ->

Analysis Settings	PRINT RESULT ANALYSIS-SETTINGS

Summary	PRINT RESULT SUMMARY
All	PRINT RESULT ALL
Importance Factors	PRINT RESULT IMPORTANCE-FACTORS
Sensitivity	PRINT RESULT SENSITIVITY
Sample	PRINT RESULT SAMPLE
Parameter Study, Main Result	PRINT RESULT PARAMETER-STUDY MAIN-RESULT
Parameter Study, Importance	PRINT RESULT PARAMETER-STUDYIMPORTANCE-FACTOR
Intermediate Results	PRINT RESULT INTERMEDIATE-RESULTS

5.1.8 The Options Menu

This menu contains the commands available in the line mode SET command, i.e. print and display settings.

Company Name	SET COMPANY-NAME
Display	SET DISPLAY
Drawing	SET DRAWING
Graph ->	
Lines and Markers	SET GRAPH LINE-OPTIONS
X Axis	SET GRAPH X-AXIS-ATTRIBUTES
Y Axis	SET GRAPH Y-AXIS-ATTRIBUTES
Z Axis	SET GRAPH Z-AXIS-ATTRIBUTES
Histogram	SET GRAPH HISTOGRAM
Pie Chart	SET GRAPH PIE-CHART
Plot	SET PLOT
Print	SET PRINT
Title	SET TITLE

5.1.9 The Help Menu

The contents of the Help menu is the same as is described with the HELP command in the next section.

5.2 Line Mode Command Syntax

This section describes the complete syntax of the line mode command input. The commands are presented alphabetically. As the line mode input is case insensitive, all alternatives are presented in upper case.

ASSIGN

ASSIGN	CONDITIONING	...
	CONTINUOUS-PROCESS	...
	CORRELATION	...
	EXTREME-VALUE	...
	FUNCTION-OPTION	...
	MEASURED-VALUE	...
	OPTIMISATION-BOUNDS	...
	SENSITIVITY-CALCULATION	...
	SIMULATION-DENSITY	...
	STARTING-POINT	...
	SUB-LEVEL-INTEGRATION	...

PURPOSE:

Assign attribute(s) to one or more named objects.

PARAMETERS:

CONDITIONING	Assign conditioning variables to a generated distribution variable.
CONTINUOUS-PROCESS	Assign duration and starting time to a type time variable and time derivative to a time dependent process variable.
CORRELATION	Assign correlation between random variables.
EXTREME-VALUE	Assign extreme value distribution type to a random variable.
FUNCTION-OPTION	Assign optional function input to a random variable that is a function of other variables, or to a model function.
MEASURED-VALUE	Assign the measured value to an event with equality constraint.
OPTIMISATION-BOUNDS	Assign bounds to a variable, limiting the range of values allowed in FORM/SORM optimization.
SENSITIVITY-CALCULATION	Assign sensitivity calculation and increment to parameters.
SIMULATION-DENSITY	Assign a variable as adjusted simulation density in a sampling of probability.

STARTING-POINT	Assign a starting point for the FORM/SORM analysis to an event.
SUB-LEVEL-INTEGRATION	Assign variables to be integrated together with time dependent process variables in a continuous process analysis.

NOTES:

None.

ASSIGN CONDITIONING

...	CONDITIONING	variable	condvar+

PURPOSE:

Assign conditioning variable(s) to a generated distribution variable or to a probability variable.

PARAMETERS:

variable	The name of a generated distribution variable or probability variable.
condvar+	A selection of variables that are kept fixed when the distribution is generated or the probability is calculated.

NOTES:

1 The current conditioning variables are presented as defaults when a generated distribution variable or a probability variable is selected.

2 The conditioning assignment to a variable is printed by use of the PRINT VARIABLE command.

See also:

- CREATE VARIABLE ... GENERATED
- CREATE VARIABLE ... PROBABILITY
- PRINT VARIABLE

EXAMPLES:

```
ASSIGN CONDITIONING GenVar ( ONLY A B C )
ASSIGN CONDITIONING PrbVar ( EXCLUDE * )
```

ASSIGN CONTINUOUS-PROCESS

...	CONTINUOUS-PROCESS	DURATION	time-variable	value
				NONE
		STARTING-TIME	time-variable	value
				NONE
		TIME-DERIVATIVE	process-variable	time-derivative-variable
				NONE

PURPOSE:

Assign duration and starting time to a time variable and assign a variable as the time derivative of a process variable

PARAMETERS:

time-variable	A variable with type attribute TIME.
process-variable	A variable with type attribute DISTRIBUTION, FITTED-DISTRIBUTION or GENERATED.
time-derivative-variable	Time derivative of process variable. A variable with type attribute DISTRIBUTION, FITTED-DISTRIBUTION or GENERATED.
value	Duration value or starting time value. Can be a numerical value or the name of a one dimensional variable.
NONE	Turn off assignment of duration value or starting point value or time derivative variable.

NOTES:

See also:

- DEFINE CONTINUOUS-PROCESS

EXAMPLES:

```
CREATE VARIABLE Time 'Time Variable'
ASSIGN CONTINUOUS-PROCESS DURATION Time DurVar
ASSIGN CONTINUOUS-PROCESS STARTING-TIME Time 0.0
CREATE VARIABLE PVar 'Process Variable' DISTRIBUTION NORMAL ...
CREATE VARIABLE TDVar 'Process Variable' DISTRIBUTION NORMAL ...
ASSIGN CONTINUOUS-PROCESS TIME-DERIVATIVE PVar TDVar
ASSIGN CONTINUOUS-PROCESS TIME-DERIVATIVE PVar NONE
```

ASSIGN CORRELATION

<table>
<tr><td rowspan="3">...</td><td rowspan="3">CORRELATION</td><td rowspan="3">univariate+</td><td>BASIC</td><td rowspan="2">value</td></tr>
<tr><td>NORMALIZED</td></tr>
<tr><td colspan="2">NONE</td></tr>
</table>

PURPOSE:

Assign the same correlation (or no correlation) to a number of variables.

PARAMETERS:

univariate+	A selection of variables that are defined as one dimensional distributions with numerical or fixed parameter values. All pairs of the selected variables will be assigned the specified correlation.
BASIC	The correlation is specified in the physical space.
NORMALIZED	The correlation is specified in the transformed standard normal space.
value	Correlation value. Can be a numerical value or the name of a one dimensional variable.

NOTES:

It is possible to do sensitivity analysis on correlation coefficients by creating them as fixed variables first, then using the fixed variable to specify the correlation value (see example below).

See also:

- PRINT CORRELATION

EXAMPLES:

```
ASSIGN CORRELATION ( P-lnC P-m ) BASIC -0.9
CREATE VARIABLE StrCorr 'Stress correlation' FIXED 0.8
ASSIGN CORRELATION ( FP-lnA FP-1dB ) NORMALIZED StrCorr
ASSIGN SENSITITIVY VARIABLE INCLUDE StrCorr
ASSIGN CORRELATION PP* NONE
```

ASSIGN EXTREME-VALUE

<table>
<tr><td rowspan="3">...</td><td rowspan="3">EXTREME-VALUE</td><td rowspan="3">variable</td><td>MIN-OF-N</td><td>n_min</td></tr>
<tr><td>MAX-OF-N</td><td>n_max</td></tr>
<tr><td colspan="2">NONE</td></tr>
</table>

PURPOSE:

Assign extreme type to a distribution variable.

PARAMETERS:

variable	A one dimensional distribution variable or a generated distribution variable.
MIN-OF-N n_min	The extreme distribution is the minimum of n_min independent, identically distributed variables with the distribution that was input when the selected variable was created/changed. n_min must be a positive whole number.
MAX-OF-N n_max	The extreme distribution is the maximum of n_max independent, identically distributed variables with the distribution that was input when the selected variable was created/changed. n_max must be a positive whole number.
NONE	No extreme type distribution is used for this variable.

NOTES:

1 All variables have by default no extreme type assigned.

2 The extreme value assignment is printed by use of the PRINT VARIABLE command.

See also:

- PRINT VARIABLE

EXAMPLES:

```
ASSIGN EXTREME-VALUE Amplitude MAX-OF-N 5
ASSIGN EXTREME-VALUE Amplitude NONE
```

ASSIGN FUNCTION-OPTION

...	FUNCTION-OPTION	FUNCTION	function	option	value
		VARIABLE	variable		

PURPOSE:

Assign input, that is not of random nature, to a model function.

PARAMETERS:

FUNCTION	Assign the value directly to a function. In this case it is applied to all variables created by use of the function (until changed again).
function	Name of the function to which the value is assigned.
VARIABLE	Assign the value to a variable that is based on a model function. This assignment affects only the selected variable, not any other variables based on the same function.
variable	Name of the variable to which the value is assigned.
option	The option to be defined. The range of available options varies from function to function.
value	The value of the option. This will be either a whole number, a floating point number, a text, a file name or a selection between alternatives, dependent on the selected option.

NOTES:

1 The default function options can be printed by use of the PRINT FUNCTION DESCRIPTION command.

2 The function options assigned to a variable are printed by use of the PRINT VARIABLE command.

3 The function options assigned to the variables created by this program should not be changed by the user.

See also:

- PRINT FUNCTION DESCRIPTION
- PRINT VARIABLE

EXAMPLES:

```
ASSIGN FUNCTION-OPTION FUNCTION F11 POWER 3
ASSIGN FUNCTION-OPTION VARIABLE VAR33 ACCURACY-TYPE RELATIVE
ASSIGN FUNCTION-OPTION VARIABLE VAR33 ACCURACY-VALUE 1.1E-5
```

ASSIGN MEASURED-VALUE

...	MEASURED-VALUE	event	variable
			NONE

PURPOSE:

Assign the measured value to an equality event

PARAMETERS:

event	The name of an event of type SINGLE.
variable	The name of the variable which was measured. This may be a coordinate in a multidimensional variable.
NONE	No measured value is assigned to the selected event.

NOTES:

1 By default no measured variable is assigned to any event, except the events describing inspections where a crack is measured to a certain size.

2 The measured value assigned to an event is printed by use of the PRINT EVENT command.

3 The measured value assignments to the events created by this program should not be changed by the user.

See also:

- PRINT EVENT

EXAMPLES:

```
ASSIGN MEASURED-VALUE FindCrack am
```

ASSIGN OPTIMISATION-BOUNDS

<table>
<tr><td rowspan="2">...</td><td rowspan="2">OPTIMISATION-BOUNDS</td><td rowspan="2">variable</td><td>MODEL-SPACE</td><td rowspan="2">...</td><td>lower</td><td>upper</td></tr>
<tr><td>U-SPACE</td><td>OFF</td><td>OFF</td></tr>
</table>

PURPOSE:

Assign bounds on variables, to be used in FORM/SORM optimization.

PARAMETERS:

variable	Name of variable to which the bounds are assigned. This is a one dimensional distribution variable, or a generated distribution variable.
MODEL-SPACE	Bounds are specified in model space (physical input values).
U-SPACE	Bounds are specified in the transformed normal space.
lower	Value of the lower bound.
upper	Value of the upper bound.
OFF	The default bound is used.

NOTES:

The optimization bounds assigned to a variable are printed by use of the PRINT VARIABLE command.

See also:

- PRINT VARIABLE

EXAMPLES:

```
ASSIGN OPTIMISATION-BOUNDS Amplitude MODEL-SPACE 0 OFF
ASSIGN OPTIMISATION-BOUNDS Load U-SPACE -20 20
```

ASSIGN SENSITIVITY-CALCULATION

...	SENSITIVITY-CALCULATION	INCREMENT	...
		VARIABLE	...

PURPOSE:

Assign sensitivity calculation parameters and increments.

PARAMETERS:

INCREMENT	Assign increment value to be used for sensitivity calculation.
VARIABLE	Select parameters for sensitivity calculation.

NOTES:

None.

ASSIGN SENSITIVITY-CALCULATION INCREMENT

...	INCREMENT	parameter	value
			DEFAULT

PURPOSE:

Assign increment to be used for sensitivity calculation.

PARAMETERS:

parameter	The parameter for which the increment applies. This can be a fixed variable, the name of a numerical parameter in a distribution variable, or the name of a numerical argument in a function variable.
value	The increment to be used.
DEFAULT	Use the default increment.

NOTES:

The specified increment overrides any increment specified by DEFINE ANALYSIS-OPTION DIFFERENTIATION.

See also:

- ASSIGN SENSITIVITY-CALCULATION VARIABLE

EXAMPLES:

```
ASSIGN SENSITIVITY-CALCULATION INCREMENT P1-lnC ON 0.01
ASSIGN SENSITIVITY-CALCULATION INCREMENT P1-lnC OFF
```

ASSIGN SENSITIVITY-CALCULATION VARIABLE

...	VARIABLE	parameter+

PURPOSE:

Select a number of parameters for sensitivity calculation.

PARAMETERS:

parameter+	The parameters to be used for sensitivity calculation. These can be a fixed variable, the name of a numerical parameter in a distribution variable, or the name of a numerical argument in a function variable.

NOTES:

1 The parameters that have previously been selected are presented as the default selection. To deassign sensitivity to some of these, remove them from the selection.

2 The command DEFINE ANALYSIS-OPTION SENSITIVITY is used to confirm or override the selection specified here.

See also:

- ASSIGN SENSITIVITY-CALCULATION INCREMENT
- DEFINE ANALYSIS-OPTION SENSITIVITY
- DEFINE FORM-SORM SENSITIVITY

EXAMPLES:

```
ASSIGN SENSITIVITY-CALCULATION VARIABLE *-Mean
ASSIGN SENSITIVITY-CALCULATION VARIABLE INCLUDE P1-lnC-Stdv
```

ASSIGN SIMULATION-DENSITY

...	SIMULATION-DENSITY	varsim	varadjsim

PURPOSE:

Assign a variable as adjusted simulation density in a sampling of probability.

PARAMETERS:

varsim	Variable representing the coordinate for which the adjusted simulation applies.
varadjsim	Variable defining the adjusted simulation density.

NOTES:

1 Adjusted simulation means that the sampling is according to the following formula in which fU is the u-space distributions and fA is the adjusted sampling density:

$$P = \int_{g(u) < 0} f_U(\boldsymbol{u}) d\boldsymbol{u} = \int_{g(u) < 0} f_A(u)(f_U(\boldsymbol{u})/f_A(u)) d\boldsymbol{u}$$

2 The adjusted simulation density replaces the variable in u-space.

3 The adjustment is restricted to Normal random variables.

4 Correlated variables and variables conditioned on the value of other variables cannot be assigned an adjusted sampling density.

See also:

- SELECT ANALYSIS-METHOD PROBABILITY DESIGN-POINT-SIMULATION ADJUSTED
- SELECT ANALYSIS-METHOD PROBABILITY MONTE-CARLO-SIMULATION ADJUSTED

EXAMPLES:

```
ASSIGN SIMULATION-DENSITY VarSim VarAdjSim
```

ASSIGN STARTING-POINT

...	STARTING-POINT	event		{	numerical	}*
		VARIABLE	variable		default	

PURPOSE:

Assign a starting point for the FORM/SORM optimization to an event or a variable.

PARAMETERS:

event	Name of the event to which the starting point is assigned. This must be a single event.
VARIABLE	Assign the starting point to a variable. This must be a one dimensional distribution variable or a generated distribution variable. This assignment causes the starting point value to be used in all events that depend on the variable, except when overridden by a direct assignment to the event.
variable	Name of the variable to which the starting point is assigned.
numerical	Numerical starting point value. The value must be specified in the physical model space, not in U-space.
default	The text default implies a default starting point value, that is the origin in U-space.

NOTES:

1 The starting point assignment can be printed by use of the PRINT STARTING-POINT command.

2 The use of starting points in the FORM/SORM optimization is determined by the DEFINE FORM-SORM STARTING-POINT INITIAL command.

3 The starting point assignment can be printed by use of the PRINT STARTING-POINT command.

4 An event may depend on several variables, and a variable may be multidimensional. It may therefore be necessary to specify more than one numerical or default value. The identification of each value (i.e. the name of the variable/parameter) is given as prompt for each needed input value.

See also:

- PRINT STARTING-POINT
- DEFINE FORM-SORM STARTING-POINT INITIAL

EXAMPLES:

```
ASSIGN STARTING-POINT EP1 7.52 DEFAULT DEFAULT 2200 -8.65
ASSIGN STARTING-POINT VARIABLE VAR7 DEFAULT DEFAULT
```

ASSIGN SUB-LEVEL-INTEGRATION

...	SUB-LEVEL-INTEGRATION	variable	ON
			OFF

PURPOSE:

Assign a sub-level-integration property to a variable.

PARAMETERS:

variable	Name of the variable to which the sub level integration property is assigned. This must be a variable with type attribute distribution or type attribute generated.
ON	The variable is pushed to the inner integration loop in a continuous process analysis.
OFF	The variable is integrated in the outer integration loop in a continuous process analysis.

NOTES:

See also:

- ASSIGN CONTINUOUS-PROCESS TIME-DERIVATIVE

EXAMPLES:

```
ASSIGN SUB-LEVEL-INTEGRATION X ON
```

CHANGE

CHANGE	EVENT	...
	FUNCTION	...
	VARIABLE	...

PURPOSE:

Change a named object.

PARAMETERS:

EVENT	Change an event.
FUNCTION	Change a function.
VARIABLE	Change a random variable.

NOTES:

None.

CHANGE EVENT

...	EVENT	name	desc	CONDITIONED	event	condition	
				INTERSECTION	subevent+		
				SINGLE	1d-variable	< , = , >	threshold
				UNION	subevent+		

PURPOSE:

To change an event.

PARAMETERS:

name	Name of event to be changed.
desc	Descriptive text for the event.
CONDITIONED	The event is a conditioned event.
event	Name of event being conditioned.
condition	Name of event conditioned on.
INTERSECTION	The event is an intersection of other events, i.e. it is fulfilled only when all subevents are fulfilled.
UNION	The event is a union of other events, i.e. it is fulfilled when at least one subevent is fulfilled.
subevent+	A selection of events forming either an intersection or a union. These cannot be conditioned events.
SINGLE	The event is a simple (in)equality.
1d-variable	Name of one dimensional variable that is the left hand side of the (in)equality.
< , = , >	One of: < less than, = equal, > greater than.
threshold	Numerical right hand side of the single event.

NOTES:

1 When the event name is selected, the existing state of the event is presented as defaults (unless the type of the event is changed).

2 The events that are created by this program should not be changed by the user.

See also:

- CREATE EVENT
- COPY EVENT
- RENAME EVENT
- DISPLAY EVENT
- PRINT EVENT
- ASSIGN STARTING-POINT
- ASSIGN MEASURED-VALUE

EXAMPLES:

```
CHANGE EVENT Loss : NPV < 0
CHANGE EVENT No1-Crack2 : INTERSECTION ( ONLY NoCrack-1 Crack2 )
CHANGE EVENT Fail-Cond 'Failure given nofind, then find' CONDITIONED Failure No1-
Crack2
```

CHANGE FUNCTION

...	FUNCTION	name	desc	FORMULA	...
				INTEGRAL	...
				RESPONSESURFACE	...

PURPOSE:

To change a function.

PARAMETERS:

name	Name of the function. Cannot be changed.
desc	Descriptive text associated with the function formula.
FORMULA	Change a function formula.
INTEGRAL	Change an integration function.
RESPONSESURFACE	Change a response surface function.

NOTES:

None.

CHANGE FUNCTION ... FORMULA

...	FORMULA	{	arguments	adesc	}*	formula-text

PURPOSE:

Change a function formula.

PARAMETERS:

argument	Name of a formula argument At least one argument must be defined.
adesc	Description of argument.
formula-text	Formula text lines.

NOTES:

Formula syntax is described in command CREATE FUNCTION FORMULA.

See also:

- CREATE FUNCTION ... FORMULA
- DELETE FUNCTION ... FORMULA
- DISPLAY FUCTION
- PRINT FUNCTION
- RENAME FUNCTION

EXAMPLES:

```
CHANGE FUNCTION FORMULA SYMFOR1 'Symbolic Formula' ( ONLY A 'Arg 1' B 'Arg 2' )
'(A+B)**2'
CHANGE FUNCTION FORMULA SYMFOR2 'Symbolic Formula' ( 'A+FUNOPT ( OPT_NAM_1 =
''Quot''''-1'', OPT_NAM_2=file.name, OPT_NAM_3=MENU_ENTRY, OPT_NAM_4 = 3,
OPT_NAM_5 = 0.5E-3,B*3-A)' )
```

CHANGE FUNCTION ... INTEGRAL

<table>
<tr><td rowspan="2">...</td><td rowspan="2">INTEGRAL</td><td rowspan="2">{</td><td rowspan="2">argname</td><td rowspan="2">argdesc</td><td rowspan="2">}*</td><td rowspan="2">function</td><td rowspan="2">{</td><td>value</td><td rowspan="2">}*</td><td rowspan="2">...</td></tr>
<tr><td>integrator</td></tr>
</table>

...	method	lowerbound	upperbound	tolerance

PURPOSE:

To change an integration function.

PARAMETERS:

argname, argdesc	Matrix of argument names and corresponding argument descriptions. At least one argument must be defined.
function	Name of function to be integrated (integrand).
value	Value can be a numerical value or an argument name, “argname”.
integrator	Text value: integrator. Case insensitive. The text value “integrator” is inserted in order to identify the single integration variable.
method	Integration method to be used. One of: ROMBERG, SIMPSON or TRAPEZOIDAL.
lowerbound	Lower bound for integrator. Must be a numerical value or an argument name, “argname”.
upperbound	Upper bound for integrator. Must be a numerical value or an argument name, “argname”.
tolerance	Relative precision in result of integration.

NOTES:

1 An argument name consists of maximum 12 alphanumeric characters and _. The first character must be alphabetic.

2 An argument description consists of maximum 50 characters.

3 The text value “integrator” may be an attribute of more than one function parameter.

See also:

- CREATE FUNCTION ... INTEGRAL
- DISPLAY FUNCTION

- PRINT FUNCTION
- RENAME FUNCTION

EXAMPLES:

Change integration of c+x from x=a to x=b to c+c+a+b+x+x from x=a to x=b

```
CHANGE FUNCTION cplusx ’Integrate c+c+a+b+x+x from x=a to x=b’ INTEGRAL ( ONLY a
’x_lower’ b ’x_upper’ c ’additive parameter’ ) Sum ( ONLY c c a b Integrator In-
tegrator ) Romberg a b 0.000001
```

CHANGE FUNCTION ... RESPONSESURFACE

...	RESPONSESURFACE	{	argname	argdesc	}*	function	...

...	{	point,argname,method,increment	}*

PURPOSE:

To change a response surface function.

PARAMETERS:

argname, argdesc	Matrix of argument names and corresponding argument descriptions. At least one argument must be defined.
function	Name of function to be approximated.
point	Centre of approximations
argname	Argument name. This approximated function argument becomes the argument argname of the approximation.
method	Function fit method to be used. L or L1: Linear approximation based on positive incrementation. L2 : Linear approximation based on two way incrementation. D : Quadratic (diagonal) approximation. No cross derivatives with other arguments. QName :Q followed by name. Quadratic approximation including cross terms for arguments that have the same group Name. Q alone is treated as a group.
increment	Increment to be used with the fit.

NOTES:

1 An argument name consists of maximum 12 alphanumeric characters and _. The first character must be alphabetic.

2 An argument description consists of maximum 50 characters.

3 Point,argname,method and increment are comma separated.

See also:

- CREATE FUNCTION ... RESPONSESURFACE
- DISPLAY FUNCTION

- PRINT FUNCTION
- PRINT RESPONSESURFACE
- RENAME FUNCTION

EXAMPLES:

Change a quadratic response surface function to appfunc centred around (1,2,3), with increment 1 for the second argument of appfunc and increment 2 for the third argument of appfunc including cross terms. The response function has two arguments, while the approximated function has three arguments:

```
CHANGE FUNCTION rspfu 'Response surface' RESPONSESURFACE ( ONLY a 'x_arg1' b
'x_arg2' ) appfunc 1 2,a,QGroup,1 3,b,QGroup,2
```

Change linear response surface function to appfunc centred around (1,2,3), with increment 1 for each argument. the second argument of appfunc and increment 2 for the third argument of appfunc including cross terms:

```
CHANGE FUNCTION rspfu 'Response surface' RESPONSESURFACE ( ONLY a 'x_arg1' b
'x_arg2' c 'x_arg3' ) appfunc 1,c,L,1 2,b,L1,1 3,a,L2,1
```

CHANGE VARIABLE

<table>
<tr><td rowspan="7">...</td><td rowspan="7">VARIABLE</td><td rowspan="7">name</td><td rowspan="7">desc</td><td>DISTRIBUTION</td><td>...</td></tr>
<tr><td>FITTED-DISTRIBUTION</td><td>...</td></tr>
<tr><td>FIXED</td><td>value</td></tr>
<tr><td>FUNCTION</td><td>...</td></tr>
<tr><td>GENERATED</td><td>1d-variable</td></tr>
<tr><td>PROBABILITY</td><td>...</td></tr>
<tr><td colspan="2">TIME</td></tr>
</table>

PURPOSE:

To change a variable.

PARAMETERS:

name	Name of variable to be changed.
desc	Descriptive text for the variable.
DISTRIBUTION	The variable is assigned a distribution. See a following page for details.
FITTED-DISTRIBUTION	The variable is assigned a distribution that is fitted to input data. See a following page for details.
FIXED	The variable has a fixed value.
value	The numerical value of a fixed variable.
FUNCTION	The variable is assigned a model function. See a following page for details.
GENERATED	The distribution of the variable is generated from the distribution of another variable.
1d-variable	The variable specifying a generated distribution. This is a one-dimensional variable or a coordinate in a multidimensional variable.
PROBABILITY	The variable is the probability of an event as calculated by Proban.
TIME	The variable is the generic time variable.

NOTES:

1 When the variable name is selected, the existing state of the variable is presented as defaults (unless the type of the variable is changed).

2 Some of the variables in a generated distribution may be shared between the generated variable and the generating variable by using the ASSIGN CONDITIONING command.

3 A (generated) distribution may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

See also:

- CREATE VARIABLE
- COPY VARIABLE
- RENAME VARIABLE
- PRINT VARIABLE
- ASSIGN CONDITIONING
- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CHANGE VARIABLE Width ' ' FIXED 22.5
CHANGE VARIABLE Amplitude ' 'GENERATED Var44
```

CHANGE VARIABLE ... DISTRIBUTION

...	DISTRIBUTION	distribution	[dim]	input-seq	parameter*
		SPLINE-1DIM	...		

PURPOSE:

To change a variable to be based on a distribution, or to change a distribution already assigned.

PARAMETERS:

distribution	The name of the distribution (excepting the spline distribution).
[dim]	The dimension of the distribution, if this is not fixed.
input-seq	The sequence of parameters used to define the distributions.
parameter	The parameter value(s) for the chosen input sequence. Each parameter value may be either a numerical value or the name of an existing one-dimensional variable. Please note that the name of a variable cannot be abbreviated here.
SPLINE-1DIM	The variable is assigned a distribution, fitted to input data. See a following page for details.

NOTES:

1 The existing values are presented as defaults whenever this is possible.

2 The variable may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

3 The distribution function and density values may be printed by use of the PRINT DISTRIBUTION command.

4 The moments of the distribution are calculated (if possible) and printed by use of the PRINT VARIABLE command.

5 The distributions are listed in SESAM User’s Manual: Proban Distributions.

See also:

- CREATE VARIABLE
- DISPLAY DISTRIBUTION
- PRINT VARIABLE
- PRINT DISTRIBUTION

- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CHANGE VARIABLE X ' ' DISTRIBUTION Normal Mean-CoV 22 0.2
CHANGE VARIABLE Y ' ' DISTRIBUTION Normal Mean-Std X 3.1
```

CHANGE VARIABLE ... DISTRIBUTION SPLINE-1DIM

<table>
<tr><td rowspan="2">...</td><td rowspan="2">SPLINE-1DIM</td><td rowspan="2">lower</td><td rowspan="2">upper</td><td>UNWEIGHTED</td><td>{</td><td>fractile</td><td>probability</td><td colspan="2">}*</td><td rowspan="2">...</td></tr>
<tr><td>WEIGHTED</td><td>{</td><td>fractile</td><td>probability</td><td>weight</td><td>}*</td></tr>
</table>

<table>
<tr><td rowspan="4">...</td><td>HIGH</td><td rowspan="4">...</td><td rowspan="3">FREE</td><td>EQUAL</td></tr>
<tr><td>MEDIUM</td><td>FREE</td></tr>
<tr><td rowspan="2">LOW</td><td>VANISH</td></tr>
<tr><td colspan="2">UNIMODAL</td></tr>
</table>

PURPOSE:

To change a variable to have a fitted distribution based on splines, or to change a spline distribution already assigned.

PARAMETERS:

lower	The lower bound of the distribution.
upper	The upper bound of the distribution.
UNWEIGHTED	Do not apply user defined weights to the spline fit.
WEIGHTED	Apply user defined weights to the input points in the spline fit.
fractile, probability	The fractiles and probability values to which the distribution function is fitted. All probabilities must be greater than 0 and smaller than 1.
fractile, probability, weighted	The fractiles and probability values to which the distribution function is fitted with corresponding weights. All probabilities must be greater than 0 and smaller than 1.
HIGH	Use high accuracy when fitting the spline distribution to the data. In some cases it may be difficult to get convergence when high accuracy is used.
MEDIUM	Use medium accuracy when fitting the spline distribution to the data.
LOW	Use low accuracy when fitting the spline distribution to the data.
FREE	The fitted distribution need not be unimodal.
UNIMODAL	The fitted distribution must be unimodal.
EQUAL	The tail values of a FREE fit must be identical.

FREE	No restriction on the tail values of a FREE fit, except that they are non-negative.
VANISH	Both tail values of a FREE fit must be zero.

NOTES:

1 The existing values are presented as defaults whenever this is possible. If changing from UNWEIGHTED to WEIGHTED, the existing fractiles and probabilities are kept as defaults, and the weights are all set to 1.

2 If the spline will not fit, try relaxing the demands on accuracy or check if any of the points have been specified wrongly.

3 The variable may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

4 The distribution function and density values may be printed by use of the PRINT DISTRIBUTION command.

5 The moments of the distribution are calculated and printed (if possible) by use of the PRINT VARIABLE command.

6 The distribution itself may be displayed using DISPLAY DISTRIBUTION. The accuracy of the fit may be examined using DISPLAY FITTED-DISTRIBUTION.

See also:

- CREATE VARIABLE
- DISPLAY DISTRIBUTION
- DISPLAY FITTED-DISTRIBUTION
- PRINT VARIABLE
- PRINT DISTRIBUTION
- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CHANGE VARIABLE X ' ' DISTRIBUTION Spline-1Dim 0 10 UNWEIGHTED ( ONLY 1.0 0.5 3.0
0.25 5.0 0.5 7.0 0.7 8.0 0.9 9.0 0.95 ) HIGH UNIMODAL
SEE ALSO:
```

CHANGE VARIABLE ... FITTED-DISTRIBUTION

...	FITTED-DISTRIBUTION	distribution	input-seq	parameter*	...

<table>
<tr><td rowspan="8">...</td><td rowspan="2">CUMULATIVE</td><td>WEIGHTED</td><td>{</td><td>Fractile</td><td>Probability</td><td>Weight</td><td>}*</td></tr>
<tr><td>UNWEIGHTED</td><td>{</td><td>Fractile</td><td>Probability</td><td colspan="2">}*</td></tr>
<tr><td rowspan="2">OBSERVATIONS</td><td>WEIGHTED</td><td>{</td><td>Observation</td><td>Weight</td><td colspan="2">}*</td></tr>
<tr><td>UNWEIGHTED</td><td></td><td>Observation</td><td colspan="3">*</td></tr>
<tr><td rowspan="2">OBSERVATION-
MOMENTFIT</td><td>WEIGHTED</td><td>{</td><td>Observation</td><td>Weight</td><td colspan="2">}*</td></tr>
<tr><td>UNWEIGHTED</td><td></td><td>Observation</td><td colspan="3">*</td></tr>
<tr><td>RESULT</td><td colspan="6">result name</td></tr>
<tr><td>RESULT-MOMENTFIT</td><td colspan="6">result name</td></tr>
</table>

PURPOSE:

To change a variable to be fitted to a distribution, or to change a fitted distribution already assigned.

PARAMETERS:

distribution	The name of the distribution (excepting the spline distribution and multidimensional distributions).
input-seq	The sequence of parameters used to define the distributions.
parameter*	The parameter specification for the chosen input sequence. Each parameter value may be either specified as a numerical value (in which case it is not fitted), as: FIT, in which case it is fitted, or as: FIT<value>, where <value> is a numerical value used as starting point for an iterative fit. A lower bound on the fitted value is specified by L<value>. An upper bound on the fitted value is specified by U<value>.
CUMULATIVE	Fit to cumulative input data.
WEIGHTED	The input data are weighted. The weights must be positive.
UNWEIGHTED	The input data are not weighted.
Fractile, Probability, Weight	Successive values of fractiles, cumulative probabilities and weights. The probabilities must be in the interval]0,1[. The input data will be sorted in order of increasing probability.
Fractile, Probability	Successive values of fractiles and cumulative probabilities. The probabilities must be in the interval]0,1[. The input data will be sorted in order of increasing probability.

OBSERVATIONS	The input data are observed values of the variable.
OBSERVATION-MOMENTFIT	The input data are observed values of the variable and first moments fit is used.
Observation, Weight	Successive values of observations and weights. The input data will be sorted in order of increasing observation values.
Observation	Observed values of the random variable to which a distribution is fitted. The input data will be sorted in order of increasing observation values.
RESULT	Fit the distribution to the results of a probability or distribution analysis. Simulation results will be fitted and stored as OBSERVATIONS, after being grouped into weighted interval data if many samples exist. Mean value based FORM results will be fitted and stored as CUMULATIVE data with equal weights on all points. Probability results from a parameter study will be fitted (if possible) and stored as CUMULATIVE data with equal weights on all points. In the case of a parameter study of a distribution analysis, the result for the first parameter value is used.
RESULT-MOMENTFIT	The input data are sampled values of the variable and first moments fit is used.
result name	The name of the result for which the distribution is to be fitted.

NOTES:

1 The existing values are presented as defaults whenever this is possible.

2 The RESULT option can be useful for substituting a variable requiring lengthy computation time with a fitted distribution.

3 The variable may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

4 The distribution function and density values may be printed by use of the PRINT DISTRIBUTION command.

5 The moments of the distribution are calculated and printed (if possible) by use of the PRINT VARIABLE command.

6 The distribution itself may be displayed using DISPLAY DISTRIBUTION. The accuracy of the fit may be examined using DISPLAY FITTED-DISTRIBUTION.

7 The distributions are listed in SESAM User's Manual: Proban Distributions.

See also:

- CREATE VARIABLE

- DISPLAY DISTRIBUTION
- DISPLAY FITTED-DISTRIBUTION
- PRINT VARIABLE
- PRINT DISTRIBUTION
- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CHANGE VARIABLE X ' ' FITTED-DISTRIBUTION Normal Mean-CoV FIT FIT OBS UNW ( ONLY
1.34 2.56 8.65 4.32 4.67 6.66 5.23 3.25 )
CHANGE VARIABLE Y ' ' FITTED-DISTRIBUTION Normal Mean-Std FIT15 FIT CUMULATIVE
WEIGHTED ( ONLY 12 0.1 1 15 0.3 2 17 0.7 1 20 0.9 1 )
CHANGE VARIABLE RES ' ' FITTED-DISTRIBUTION Lognormal Mean-Std-L FIT FIT 0 RESULT
LastAnalysis
```

CHANGE VARIABLE ... FUNCTION

...	FUNCTION	function	[dim]	argument*

PURPOSE:

To change a variable to be based on a model function, or to change a function already assigned.

PARAMETERS:

function	The name of the function. The functions can be listed by use of the commands PRINT FUNCTION LIBRARY and PRINT FUNCTION DESCRIPTION.
[dim]	The dimension of the function, if this is not fixed.
argument*	The argument value(s) for the chosen function. Each argument value may be either a numerical value or the name of an existing one-dimensional variable. Please note that the name of a variable cannot be abbreviated here.

NOTES:

1 The existing values are presented as defaults whenever this is possible.

2 The variables that are created by this program should not be changed by the user.

3 The selection of functions presented is determined by the current selection of sub-libraries (see SELECT FUNCTION-LIBRARY). This is because some libraries may contain a large number of functions and/or not be relevant to the current problem.

See also:

- CREATE VARIABLE
- PRINT VARIABLE
- PRINT FUNCTION
- SELECT FUNCTION-LIBRARY

EXAMPLES:

```
CHANGE VARIABLE Total-Durati ' ' FUNCTION Sum ( EXCLUDE Path-1 )
CHANGE VARIABLE Diff1 ' ' FUNCTION Difference Resist5 Load4
```

CHANGE VARIABLE ... PROBABILITY

<table>
<tr><td rowspan="3">...</td><td rowspan="3">PROBABILITY</td><td>RELIABILITY-INDEX</td><td rowspan="3">event</td></tr>
<tr><td>PROBABILITY</td></tr>
<tr><td>LOG-PROBABILITY</td></tr>
</table>

PURPOSE:

To change a variable to have the probability of an event as value.

PARAMETERS:

RELIABILITY-INDEX	The reliability index corresponding to the probability of event.
PROBABILITY	The probability of event.
LOG-PROBABILITY	The natural logarithm of the probability of event.
event	The name of an existing event

NOTES:

1 <event> must not be a conditional event or contain equality events.

2 <event> must be calculable by using FORM.

3 The probability of <event> may be calculated with the value of variables in the model for <event> as arguments. The selection of variables is made by using the command ASSIGN CONDITIONING.

See also:

- CREATE VARIABLE
- COPY VARIABLE
- RENAME VARIABLE
- PRINT VARIABLE
- ASSIGN CONDITIONING

EXAMPLES:

```
CHANGE VARIABLE P_EVENT ' ' PROBABILITY RELIABILITY-INDEX EVENAM
```

COPY EVENT

...	EVENT	from	to

PURPOSE:

To copy one event to another.

PARAMETERS:

from	Name of the event to be copied.
to	Name of the new event. This cannot be the name of an existing event.

NOTES:

Only the basic contents of the event (i.e. those defined in CREATE) are copied. Assignments are not copied.

See also:

- CHANGE EVENT
- CREATE EVENT
- DELETE EVENT
- RENAME EVENT
- PRINT EVENT
- DISPLAY EVENT
- ASSIGN MEASURED-VALUE
- ASSIGN STARTING-POINT

EXAMPLES:

```
COPY EVENT Moment-1 Moment-2
```

COPY VARIABLE

...	VARIABLE	from	to

PURPOSE:

To copy one variable to another.

PARAMETERS:

from	The name of the variable to be copied.
to	The name of the new variable. This cannot be the name of an existing variable.

NOTES:

Only the basic contents of the event (i.e. those defined in CREATE) are copied. Assignments are not copied.

See also:

- CHANGE VARIABLE
- CREATE VARIABLE
- DELETE VARIABLE
- RENAME VARIABLE
- PRINT VARIABLE
- DISPLAY VARIABLE
- ASSIGN CONDITIONING
- ASSIGN CORRELATION
- ASSIGN EXTREME-VALUE
- ASSIGN FUNCTION-OPTION
- ASSIGN OPTIMISATION-BOUNDS
- ASSIGN SENSITIVITY-CALCULATION
- ASSIGN STARTING-POINT

EXAMPLES:

```
COPY VARIABLE Width1 Width2
```

CREATE

<table>
<tr><td rowspan="4">CREATE</td><td>EVENT</td><td colspan="2">...</td></tr>
<tr><td rowspan="2">FUNCTION</td><td>FORMULA</td><td>...</td></tr>
<tr><td>INTEGRATION</td><td>...</td></tr>
<tr><td>VARIABLE</td><td colspan="2">...</td></tr>
</table>

PURPOSE:

Create a named object.

PARAMETERS:

EVENT	Create an event.
FUNCTION	Create a function.
VARIABLE	Create a random variable.

NOTES:

None.

CREATE EVENT

<table>
<tr><td rowspan="4">...</td><td rowspan="4">EVENT</td><td rowspan="4">name</td><td rowspan="4">desc</td><td>CONDITIONED</td><td>event</td><td colspan="2">condition</td></tr>
<tr><td>INTERSECTION</td><td colspan="3">subevent+</td></tr>
<tr><td>SINGLE</td><td>1d-variable</td><td>< , = , ></td><td>threshold</td></tr>
<tr><td>UNION</td><td colspan="3">subevent+</td></tr>
</table>

PURPOSE:

To create an event.

PARAMETERS:

name	Name of event. This cannot be the name of an existing event. Event names are matched case insensitive and can not be longer than 12 characters.
desc	Descriptive text for the event. It can be up to 50 characters long.
CONDITIONED	Event is a conditioned event.
event	Name of event being conditioned.
condition	Name of event conditioned on.
INTERSECTION	Event is an intersection of other events, i.e. it is fulfilled only when all subevents are fulfilled.
UNION	Event is a union of other events, i.e. it is fulfilled when at least one subevent is fulfilled.
subevent+	Selection of events forming either an intersection of union. These cannot be conditioned events.
SINGLE	Event is a simple (in)equality.
1d-variable	Name of the one dimensional variable that is forming the left hand side if the (in)equality.
< , = , >	One of: < less than, = equal, > greater than.
threshold	Numerical right hand side of the single event.

NOTES:

See also:

- CHANGE EVENT

- COPY EVENT
- RENAME EVENT
- DELETE EVENT
- DISPLAY EVENT
- PRINT EVENT
- ASSIGN STARTING-POINT
- ASSIGN MEASURED-VALUE

EXAMPLES:

```
CREATE EVENT Loss 'Negative net present value' NPV < 0
CREATE EVENT No1-Crack2 'Both inspections' INTERSECTION ( ONLY NoCrack-1 Crack2 )
CREATE EVENT Fail-Cond 'Failure given nofind, then find' CONDITIONED Failure No1-
Crack2
```

CREATE FUNCTION

<table>
<tr><td rowspan="3">...</td><td rowspan="3">FUNCTION</td><td rowspan="3">name</td><td rowspan="3">desc</td><td>FORMULA</td><td>...</td></tr>
<tr><td>INTEGRAL</td><td>...</td></tr>
<tr><td>RESPONSESURFACE</td><td>...</td></tr>
</table>

PURPOSE:

To create a function.

PARAMETERS:

name	Name of the function. This name must be unique among functions and no longer than 12 characters. Names are matched case insensitive.
desc	Descriptive text associated with the function formula.
FORMULA	Create a function formula.
INTEGRAL	Create an integration function.
RESPONSESURFACE	Create a response surface function.

NOTES:

None.

CREATE FUNCTION ... FORMULA

...	FORMULA	{	argname	argdesc	}*	formula-text

PURPOSE:

To create a function formula.

PARAMETERS:

argname, argdesc	A matrix of argument names and corresponding argument descriptions. At least one argument must be defined.
formula-text	Formula text lines.

NOTES:

1 An argument name consists of maximum 12 alphanumeric characters and _. The first character must be alphabetic.

2 An argument description consists of maximum 50 characters.

3 A formula is input through a number of lines that are concatenated. The order of calculation is according to the FORTRAN syntax. See the syntax below.

Unary operators:	+ plus sign, - minus sign	
Binary operators:	+ addition , - subtraction, * multiplication, / division, ** exponentiation	
Separator:	, separates the elements of a function argument/option list	
Delimiters:	(left parenthesis,) right parenthesis. Delimits a function argument/option list and a portion of a formula.	
Operators...	Association	Precedence
FUNAM(,), ()	left to right	5
**	right to left	4
Unary +, Unary -	right to left	3
*,/	left to right	2
binary +, binary -	left to right	1
Quotes:'	delimits a character value. An apostrophe within a quoted text must be entered as '' in graphics mode and as '''' in line mode/on journal file.	
Blanks	Blanks are deleted except within quoted texts.	
Hyphen -:	A hyphen - in the defined name for a function, function option or function option menu entry, must be entered as _. Names should be unique when - is replaced by _.	

Case sensitivity:	The formula text is case insensitive, except within a quoted string (function option value).
Function option:	A function option is entered as OPTION_NAME=OPTION-VALUE.

See also:

- CHANGE FUNCTION
- DISPLAY FUNCTION
- PRINT FUNCTION
- RENAME FUNCTION

EXAMPLES:

```
CREATE FUNCTION SYMFOR1 'Symbolic formula' FORMULA ( ONLY A 'Arg A' B 'Arg B' )
'(A+B)**2'
CREATE FUNCTION SYMFOR2 'Symbolic formula' FORMULA ( 'A+FUNOPT( OPT_NAM_1 ='
'Quot''''-1'', OPT_NAM_2 = file.name, OPT_NAM_3 = MENU_ENTRY, OPT_NAM_4 = 3,
OPT_NAM_5 = 0.5E-3, B*3-A)' )
```

CREATE FUNCTION ... INTEGRAL

<table>
<tr><td rowspan="2">...</td><td rowspan="2">INTEGRAL</td><td rowspan="2">{</td><td rowspan="2">argname</td><td rowspan="2">argdesc</td><td rowspan="2">}*</td><td rowspan="2">function</td><td rowspan="2">{</td><td>value</td><td rowspan="2">}*</td><td rowspan="2">...</td></tr>
<tr><td>integrator</td></tr>
</table>

...	method	lowerbound	upperbound	tolerance

PURPOSE:

To create an integration function.

PARAMETERS:

argname, argdesc	Matrix of argument names and corresponding argument descriptions. At least one argument must be defined.
function	Name of function to be integrated (integrand).
value	Value can be a numerical value or an input argument name, “argname”.
integrator	Text value: integrator. Case insensitive. The text value “integrator” is inserted in order to identify the single integration variable.
method	Integration method to be used. One of: ROMBERG, SIMPSON or TRAPEZOIDAL.
lowerbound	Lower bound for integrator. Must be a numerical value or an argument name, “argname”.
upperbound	Upper bound for integrator. Must be a numerical value or an argument name, “argname”
tolerance	Relative precision in result of integration.

NOTES:

1 An argument name consists of maximum 12 alphanumeric characters and _. The first character must be alphabetic.

2 An argument description consists of maximum 50 characters.

3 The text value “integrator” may be an attribute of more than one function parameter.

See also:

- CHANGE FUNCTION ... INTEGRAL
- DISPLAY FUNCTION

- PRINT FUNCTION
- RENAME FUNCTION

EXAMPLES:

Integrate c+x from x=a to x=b:

```
CREATE FUNCTION cplusx ’Integrate c+x from from a to b’ INTEGRAL ( ONLY a ’x_lower’
b ’x_upper’ c ’additive parameter’ ) SUM ( ONLY c Integrator ) Romberg a b 0.000001
```

CREATE FUNCTION ... RESPONSESURFACE

...	RESPONSESURFACE	{	argname	argdesc	}*	function	...

...	{	point,argname,method,increment	}*

PURPOSE:

To create a response surface function.

PARAMETERS:

argname, argdesc	Matrix of argument names and corresponding argument descriptions. At least one argument must be defined.
function	Name of function to be approximated.
point	Centre of approximations
argname	Argument name. This approximated function argument becomes the argument argname of the approximation.
method	Function fit method to be used. L or L1: Linear approximation based on positive incrementation. L2 : Linear approximation based on two way incrementation. D : Quadratic (diagonal) approximation. No cross derivatives with other arguments. QName :Q followed by name. Quadratic approximation including cross terms for arguments that have the same group Name. Q alone is treated as a group.
increment	Increment to be used with the fit.

NOTES:

1 An argument name consists of maximum 12 alphanumeric characters and _. The first character must be alphabetic.

2 An argument description consists of maximum 50 characters.

3 Point,argname,method and increment are comma separated.

See also:

- CHANGE FUNCTION ... RESPONSESURFACE
- DISPLAY FUNCTION

- PRINT FUNCTION
- PRINT RESPONSESURFACE
- RENAME FUNCTION

EXAMPLES:

Fit a quadratic response surface function to appfunc centred around (1,2,3), with increment 1 for the second argument of appfunc and increment 2 for the third argument of appfunc including cross terms. The response function has two arguments, while the approximated function has three arguments:

```
CREATE FUNCTION rspfu ’Response surface’ RESPONSESURFACE ( ONLY a ’x_arg1’ b
’x_arg2’ ) appfunc 1 2,a,QGroup,1 3,b,QGroup,2
```

Fit linear response surface function to appfunc centred around (1,2,3), with increment 1 for each argument. the second argument of appfunc and increment 2 for the third argument of appfunc including cross terms:

```
CREATE FUNCTION rspfu ’Response surface’ RESPONSESURFACE ( ONLY a ’x_arg1’ b
’x_arg2’ c ’x_arg3’ ) appfunc 1,c,L,1 2,b,L1,1 3,a,L2,1
```

CREATE VARIABLE

<table>
<tr><td rowspan="7">...</td><td rowspan="7">VARIABLE</td><td rowspan="7">name</td><td rowspan="7">desc</td><td>DISTRIBUTION</td><td>...</td></tr>
<tr><td>FITTED-DISTRIBUTION</td><td>...</td></tr>
<tr><td>FIXED</td><td>value</td></tr>
<tr><td>FUNCTION</td><td>...</td></tr>
<tr><td>GENERATED</td><td>1d-variable</td></tr>
<tr><td>PROBABILITY</td><td>...</td></tr>
<tr><td colspan="2">TIME</td></tr>
</table>

PURPOSE:

To create a variable

PARAMETERS:

name	Name of the variable to be created. This cannot be the name of an existing variable. Variable names are matched case insensitive and can be up to 12 characters long.
desc	Descriptive text for the variable. It can be up to 50 characters long.
DISTRIBUTION	Variable is assigned a distribution. See a following page for details.
FITTED-DISTRIBUTION	Variable is assigned a distribution that is fitted to input data. See a following page for details.
FIXED	Variable has a fixed value.
value	Numerical value of a fixed variable.
FUNCTION	Variable assigned a model function. See a following page for details.
GENERATED	The distribution of the variable is generated from the distribution of another variable.
1d-variable	Variable specifying a generated distribution. This is a one-dimensional variable or a coordinate in a multidimensional variable.
PROBABILITY	The variable is the probability of an event.
TIME	The variable is generic time.

NOTES:

1 Some of the variables in a generated distribution may be shared between the generated variable and the generating variable by using the ASSIGN CONDITIONING command.

2 A generated distribution may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

See also:

- CHANGE VARIABLE
- COPY VARIABLE
- DELETE VARIABLE
- RENAME VARIABLE
- PRINT VARIABLE
- ASSIGN CONDITIONING
- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CREATE VARIABLE Width 'Width of plate' FIXED 22.5
CREATE VARIABLE Amplitude 'Wave amplitude' GENERATED Var44
```

CREATE VARIABLE ... DISTRIBUTION

...	DISTRIBUTION	distribution	[dim]	input-seq	parameter*
		SPLINE-1DIM	...		

PURPOSE:

To create a variable to be based on a distribution.

PARAMETERS:

distribution	Name of the distribution (excepting the spline distribution).
[<dim>]	The dimension of the distribution, if this is not fixed.
input-seq	The sequence of parameters used to define the distributions.
parameter*	The parameter value(s) for the chosen input sequence. Each parameter value may be either a numerical value or the name of an existing one-dimensional variable. Please note that the name of a variable cannot be abbreviated here.
SPLINE-1DIM	The variable is assigned a distribution, fitted to input data. See a following page for details.

NOTES:

1 The variable may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

2 The distribution function and density values may be printed by use of the PRINT DISTRIBUTION command.

3 The moments of the distribution are calculated and printed (if possible) by use of the PRINT VARIABLE command.

4 The distributions are listed in SESAM User’s Manual: Proban Distributions.

See also:

- CHANGE VARIABLE
- DISPLAY DISTRIBUTION
- PRINT VARIABLE
- PRINT DISTRIBUTION
- ASSIGN CORRELATION

- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CREATE VARIABLE X ' ' DISTRIBUTION Normal Mean-CoV 22 0.2
CREATE VARIABLE Y ' ' DISTRIBUTION Normal Mean-Std X 3.1
```

CREATE VARIABLE ... DISTRIBUTION SPLINE-1DIM

<table>
<tr><td rowspan="2">...</td><td rowspan="2">SPLINE-1DIM</td><td rowspan="2">lower</td><td rowspan="2">upper</td><td>UNWEIGHTED</td><td>{</td><td>fractile</td><td>probability</td><td>}*</td><td colspan="2">...</td></tr>
<tr><td>WEIGHTED</td><td>{</td><td>fractile</td><td>probability</td><td>weight</td><td>}*</td><td>...</td></tr>
</table>

<table>
<tr><td rowspan="4">...</td><td>HIGH</td><td rowspan="4">...</td><td rowspan="3">FREE</td><td>EQUAL</td></tr>
<tr><td>MEDIUM</td><td>FREE</td></tr>
<tr><td rowspan="2">LOW</td><td>VANISH</td></tr>
<tr><td colspan="2">UNIMODAL</td></tr>
</table>

PURPOSE:

To create a variable to have a fitted distribution based on splines.

PARAMETERS:

lower	The lower bound of the distribution.
upper	The upper bound of the distribution.
UNWEIGHTED	Do not apply user defined weights to the spline fit.
WEIGHTED	Apply user defined weights to the input points in the spline fit.
fractile, probability	The fractiles and probability values to which the distribution function is fitted. All probabilities must be greater than 0 and smaller than 1.
fractile, probability, weighted	The fractiles and probability values to which the distribution function is fitted with corresponding weights. All probabilities must be greater than 0 and smaller than 1.
HIGH	Use high accuracy when fitting the spline distribution to the data. In some cases it may be difficult to get convergence when high accuracy is used.
MEDIUM	Use medium accuracy when fitting the spline distribution to the data.
LOW	Use low accuracy when fitting the spline distribution to the data.
FREE	The fitted distribution need not be unimodal.
UNIMODAL	The fitted distribution must be unimodal.
EQUAL	The tail values of a FREE fit must be identical.

FREE	No restriction on the tail values of a FREE fit, except that they are non-negative.
VANISH	Both tail values of a FREE fit must be zero.

NOTES:

1 If the spline will not fit, try relaxing the demands on accuracy or check if any of the points have been specified wrongly.

2 The variable may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

3 The distribution function and density values may be printed by use of the PRINT DISTRIBUTION command.

4 The moments of the distribution are calculated and printed (if possible) by use of the PRINT VARIABLE command.

5 The distribution itself may be displayed using DISPLAY DISTRIBUTION. The accuracy of the fit may be examined using DISPLAY FITTED-DISTRIBUTION.

See also:

- CHANGE VARIABLE
- DISPLAY DISTRIBUTION
- PRINT VARIABLE
- DISPLAY FITTED-DISTRIBUTION
- PRINT DISTRIBUTION
- ASSIGN CORRELATION
- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CREATE VARIABLE X ' ' DISTRIBUTION Spline-1Dim 0 10 UNWEIGHTED ( ONLY 1.0 0.5 3.0
0.25 5.0 0.5 7.0 0.7 8.0 0.9 9.0 0.95 ) HIGH UNIMODAL
```

CREATE VARIABLE ... FITTED-DISTRIBUTION

...	FITTED-DISTRIBUTION	distribution	input-seq	parameter*	...

...	CUMULATIVE	WEIGHTED	{	fractile	probability	weight	}*
		UNWEIGHTED	{	fractile	probability	}*	
	OBSERVATIONS	WEIGHTED	{	observation	weight	}*	
		UNWEIGHTED	{	observation	}*		
	OBSERVATION-MOMENTFIT	WEIGHTED	{	observation	weight	}*	
		UNWEIGHTED	{	observation	}*		
	RESULT	result name					
	RESULT-MOMENTFIT	result name					

PURPOSE:

To create a variable to be fitted to a distribution.

PARAMETERS:

distribution	The name of the distribution (excepting the spline distribution and multidimensional distributions).
input-seq	The sequence of parameters used to define the distributions.
parameter*	The parameter specification for the chosen input sequence. Each parameter value may be either specified as a numerical value (in which case it is not fitted), as: FIT, in which case it is fitted, or as: FIT<value>, where <value> is a numerical value used as starting point for an iterative fit. A lower bound on the fitted value is specified by L<value>. An upper bound on the fitted value is specified by U<value>.
CUMULATIVE	Fit to cumulative input data.
WEIGHTED	The input data are weighted. The weights must be positive.
UNWEIGHTED	The input data are not weighted.
fractile, probability, weight	Successive values of fractiles, cumulative probabilities and weights. The probabilities must be in the interval]0,1[. The input data will be sorted in order of increasing probability.

fractile, probability	Successive values of fractiles and cumulative probabilities. The probabilities must be in the interval]0,1[. The input data will be sorted in order of increasing probability.
OBSERVATION-MOMENTFIT	The input data are observed values of the variable and first moments fit is used.
observation, weight	Successive values of observations and weights. The input data will be sorted in order of increasing observation values.
observation	Observed values of the random variable to which a distribution is fitted. The input data will be sorted in order of increasing observation values.
RESULT-MOMENTFIT	The input data are sampled values of the variable and first moments fit is used.
result name	The name of the result for which the distribution is to be fitted.

NOTES:

1 The existing values are presented as defaults whenever this is possible.

2 The RESULT option can be useful for substituting a variable requiring lengthy computation time with a fitted distribution.

3 The variable may be assigned an extreme type distribution by using the ASSIGN EXTREME-VALUE command.

4 The distribution function and density values may be printed by use of the PRINT DISTRIBUTION command.

5 The moments of the distribution are calculated and printed (if possible) by use of the PRINT VARIABLE command.

6 The distribution itself may be displayed using DISPLAY DISTRIBUTION. The accuracy of the fit may be examined using DISPLAY FITTED-DISTRIBUTION.

7 The distributions are listed in SESAM User's Manual: Proban Distributions.

See also:

- CREATE VARIABLE
- DISPLAY DISTRIBUTION
- DISPLAY FITTED-DISTRIBUTION
- PRINT VARIABLE
- PRINT DISTRIBUTION

- ASSIGN EXTREME-VALUE

EXAMPLES:

```
CREATE VARIABLE X ' ' FITTED-DISTRIBUTION Normal Mean-CoV FIT FIT OBS UNW ( ONLY
1.34 2.56 8.65 4.32 4.67 6.66 5.23 3.25 )
CREATE VARIABLE Y ' ' FITTED-DISTRIBUTION Normal Mean-Std FIT15 FIT CUMULATIVE
WEIGHTED ( ONLY 12 0.1 1 15 0.3 2 17 0.7 1 20 0.9 1 )
CREATE VARIABLE RES ' ' FITTED-DISTRIBUTION Lognormal Mean-Std-L FIT FIT 0 RESULT
LastAnalysis
```

CREATE VARIABLE ... FUNCTION

...	FUNCTION	function	[dim]	argument*

PURPOSE:

To create a variable to be a function of numerical values or other variables.

PARAMETERS:

function	The name of the function. The functions can be listed by use of the commands PRINT FUNCTION LIBRARY and PRINT FUNCTION DESCRIPTION.
[dim]	The dimension of the function, if this is not fixed.
argument*	The argument value(s) for the chosen function. Each argument value may be either a numerical value or the name of an existing one-dimensional variable. Please note that the name of a variable cannot be abbreviated here.

NOTES:

The selection of functions presented is determined by the current selection of sub-libraries (see SELECT FUNCTION-LIBRARY). This is because some libraries may contain a large number of functions and/or not be relevant to the current problem.

See also:

- CHANGE VARIABLE
- PRINT VARIABLE
- PRINT FUNCTION
- SELECT FUNCTION-LIBRARY
- ASSIGN WAVE-DIRECTION-PROBABILITY
- CHANGE WAVE-STATISTICS
- PRINT WAVE-STATISTICS
- ASSIGN MODEL-FACTOR SCATTER-DISTRIBUTION
- ASSIGN MODEL-FACTOR WAVE-SPECTRUM-SHAPE
- ASSIGN MODEL-FACTOR WAVE-SPREADING
- ASSIGN UNCERTAINTY VALUE

EXAMPLES:

```
CREATE VARIABLE Total-Durati 'Total duration of project' FUNCTION Sum ( EXCLUDE
Path-1 )
CREATE VARIABLE Diff1 ' ' FUNCTION Difference Resist5 Load4
```

CREATE VARIABLE ... PROBABILITY

...	PROBABILITY	RELIABILITY-INDEX	event
		PROBABILITY	
		LOG-PROBABILITY	

PURPOSE:

To create a variable to have the probability of an event as value.

PARAMETERS:

RELIABILITY-INDEX	Reliability index corresponding to the probability of event.
PROBABILITY	Probability of event.
LOG-PROBABILITY	Natural logarithm of the probability of event.
event	Name of an existing event.

NOTES:

1 Event must not be a conditional event or contain equality events.

2 Event must be calculable by using FORM.

3 The probability of event may be calculated with the value of variables in the model for event as arguments. The selection of variables is made by using the command ASSIGN CONDITIONING.

See also:

- CHANGE VARIABLE
- COPY VARIABLE
- RENAME VARIABLE
- PRINT VARIABLE
- ASSIGN CONDITIONING

EXAMPLES:

```
CREATE VARIABLE P_EVENT ' ' PROBABILITY RELIABILITY-INDEX EVENAM
```

DEFINE

DEFINE	ANALYSIS-OPTION	...
	CONTINUOUS-PROCESS	...
	DISTRIBUTION-SIMULATION	...
	FORM-SORM	...
	MEAN-VALUE-FORM	...
	PARAMETER-STUDY	...
	PRESENTATION	...
	PROBABILITY-SIMULATION	...
	TWO-PARAMETER-STUDY	...

PURPOSE:

Define global parameters or analysis options.

PARAMETERS:

ANALYSIS-OPTION	Define general options for distribution and probability analysis.
CONTINUOUS-PROCESS	Define general options for crossing rate and first passage analysis.
DISTRIBUTION-SIMULATION	Define simulation of distributions.
FORM-SORM	Define options for FORM and SORM probability analysis.
MEAN-VALUE-FORM	Define how a mean based FORM distribution analysis is performed.
PARAMETER-STUDY	Define values of a parameter for repeated analysis as a function of this parameter.
PRESENTATION	Define options used for presentation (print and display).
PROBABILITY-SIMULATION	Define options for simulation of probabilities.
PARAMETER-STUDY	Define for each of two parameters an array of values for repeated analysis covering the matrix of values defined by the two arrays.

DEFINE ANALYSIS-OPTION

<table>
<tr><td rowspan="29">...</td><td rowspan="29">ANALYSIS-OPTION</td><td>DIFFERENTIATION</td><td>uspace1</td><td>uspace2</td><td>rel</td><td>abs</td><td>limit</td></tr>
<tr><td>GENERATED-DISTRIBUTION</td><td colspan="5">...</td></tr>
<tr><td rowspan="4">GRADIENT-CALCULATION</td><td colspan="5">ANALYTICAL</td></tr>
<tr><td colspan="5">ONEWAY-INCREMENTATION</td></tr>
<tr><td colspan="5">TWOWAY-INCREMENTATION</td></tr>
<tr><td colspan="5">NUMERICAL</td></tr>
<tr><td>U-SPACE-BOUNDS</td><td colspan="5">Value</td></tr>
<tr><td rowspan="2">IMPORTANCE-FACTORS</td><td colspan="5">ON</td></tr>
<tr><td colspan="5">OFF</td></tr>
<tr><td rowspan="10">INTERMEDIATE-RESULTS</td><td colspan="3" rowspan="2">GRADIENT-VALUES</td><td colspan="2">ON</td></tr>
<tr><td colspan="2">OFF</td></tr>
<tr><td rowspan="4">LEVEL</td><td colspan="4">NONE</td></tr>
<tr><td colspan="4">LOW</td></tr>
<tr><td colspan="4">MEDIUM</td></tr>
<tr><td colspan="4">EXCESSIVE</td></tr>
<tr><td colspan="2" rowspan="2">POINT-VALUES</td><td colspan="3">ON</td></tr>
<tr><td colspan="3">OFF</td></tr>
<tr><td colspan="4" rowspan="2">SHOW-DURING-ANALYSIS</td><td>ON</td></tr>
<tr><td>OFF</td></tr>
<tr><td>NESTED-ANALYSIS</td><td colspan="5">...</td></tr>
<tr><td rowspan="2">PARAMETER-STUDY</td><td colspan="5">ON</td></tr>
<tr><td colspan="5">OFF</td></tr>
<tr><td rowspan="3">SEEDS</td><td colspan="5">DEFAULT</td></tr>
<tr><td colspan="5">RANDOM</td></tr>
<tr><td>seed1</td><td>seed2</td><td colspan="3">seed3</td></tr>
<tr><td rowspan="3">SENSITIVITY</td><td colspan="5">ALL</td></tr>
<tr><td colspan="5">NONE</td></tr>
<tr><td colspan="5">SELECTED</td></tr>
</table>

PURPOSE:

Define analysis options for probability and distribution analyses.

PARAMETERS:

DIFFERENTIATION	Define differentiation increments for use in FORM/SORM optimization and in calculation of sensitivity values.
uspace1	The differentiation increment in U-space. It must be positive.
uspace2	The differentiation increment for the Hessian matrix in U-space. Used during the FORM/SORM optimization. It must be positive.
rel	Relative parameter increment. It must be positive.
abs	Absolute parameter increment. It must be positive.
limit	Limit for application of relative parameter increment. The absolute increment is used if the absolute value of the parameter is less than limit. It must be positive.
GENERATED-DISTRIBUTION	Define analysis options for use of generated distributions. See a following page.
GRADIENT-CALCULATION	Determines if the gradients that have been programmed into the model functions are used (ANALYTICAL), or if one way (u+du) or two way (u+du and u-du) incrementation is used to determine the gradient. NUMERICAL is obsolete, but points to one way incrementation.
U-SPACE-BOUNDS	Initialises the u-space optimisation upper bounds to Value and the u-space lower bounds to -Value.
IMPORTANCE-FACTORS	Controls if importance factors are calculated. ON/OFF.
INTERMEDIATE-RESULTS	During an analysis, intermediate results may be stored on the database and possibly written to the screen. This is mainly in order to facilitate debugging of the probabilistic model.
GRADIENT-VALUES	Controls if gradient values are shown during the analysis.
LEVEL	Controls the amount of intermediate results to be generated. The possible alternatives are: NONE, LOW, MEDIUM, EXCESSIVE.
POINT/VALUES	Controls if point values (e.g. values of variables forming single events) are shown during the analysis.

SHOW-DURING-ANALYSIS	Controls whether the immediate results will be shown on the screen during the analysis run. Please take care, as excessive amounts of output may be generated.
NESTED-ANALYSIS	Define analysis options for use of probability variables. See a following page.
PARAMETER-STUDY	Controls if an assigned parameter study is actually performed.
SEEDS	Controls specification of seeds for the pseudo-random number generator. The generator requires three integer seeds. If two otherwise identical simulations are started with the same seeds. they will produce the same results.
DEFAULT	The default seeds are: 699570728 398267609 1044576128. These are mostly useful for testing (reproduction of results).
RANDOM	The seeds are generated randomly from the date and time. This works quite well, and is recommended for most simulations.
seed1, seed2, eed3	A direct specification of the three integer seeds.
SENSITIVITY	Controls the extent of the parametric sensitivity calculation (does not control importance factor calculation). May be used to override the assignments done by the ASSIGN SENSITIVITY-CALCULATION command. The possible alternatives are: ALL (calculate all), SELECTED (calculate assigned values) or NONE.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- DEFINE PARAMETER-STUDY
- ASSIGN SENSITIVITY-CALCULATION
- PRINT ANALYSIS-SETTINGS

EXAMPLE:

The following values are default when the program starts up with a new database:

```
DEFINE ANALYSIS-OPTION DIFFERENTIATION 0.001 0.1 0.0001 0.001 1.0E-10
DEFINE ANALYSIS-OPTION GRADIENT-CALCULATION ANALYTICAL
DEFINE ANALYSIS-OPTION IMPORTANCE-FACTORS ON
DEFINE ANALYSIS-OPTION INTERMEDIATE RESULTS GRADIENT-VALUES OFF
DEFINE ANALYSIS-OPTION INTERMEDIATE RESULTS LEVEL NONE
DEFINE ANALYSIS-OPTION INTERMEDIATE RESULTS POINT-VALUES OFF
DEFINE ANALYSIS-OPTION INTERMEDIATE RESULTS SHOW-DURING-ANALYSIS OFF
DEFINE ANALYSIS-OPTION PARAMETER-STUDY ON
```

```
DEFINE ANALYSIS-OPTION SEEDS RANDOM
DEFINE ANALYSIS-OPTION SENSITIVITY SELECTED
```

DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION

...	GENERATED-DISTRIBUTION	...

<table>
<tr><td rowspan="14">...</td><td>DIFFERENTIATION</td><td>uspace1</td><td>uspace2</td><td>rel</td><td>abs</td><td>limit</td></tr>
<tr><td rowspan="3">GRADIENT-CALCULATION</td><td colspan="5">ANALYTICAL</td></tr>
<tr><td colspan="5">ONEWAY-INCREMENTATION</td></tr>
<tr><td colspan="5">TWOWAY-INCREMENTATION</td></tr>
<tr><td>U-SPACE-BOUNDS</td><td colspan="5">Value</td></tr>
<tr><td rowspan="4">INTERMEDIATE-RESULTS</td><td colspan="5">NONE</td></tr>
<tr><td colspan="5">LOW</td></tr>
<tr><td colspan="5">MEDIUM</td></tr>
<tr><td colspan="5">EXCESSIVE</td></tr>
<tr><td>FRACTILE-FROM-PROBABILITY</td><td>UNMIN</td><td>maxit</td><td>maxstep</td><td colspan="2">conv</td></tr>
<tr><td rowspan="3">PROBABILITY-FROM-FRACTILE</td><td>SQP</td><td>maxit</td><td>maxstep</td><td colspan="2">conv</td></tr>
<tr><td>NLPQL</td><td colspan="4">...</td></tr>
<tr><td>RFCRC</td><td colspan="4">...</td></tr>
</table>

PURPOSE:

Define analysis options for usage of generated distributions.

PARAMETERS:

DIFFERENTIATION	Define differentiation increments for use in optimization.
uspace1	The differentiation increment in U-space. It must be positive.
uspace2	The differentiation increment for the Hessian matrix in U-space. Used during the FORM/SORM optimization. It must be positive.
rel	Relative parameter increment. It must be positive.
abs	Absolute parameter increment. It must be positive.
limit	Limit for application of relative parameter increment. The absolute increment is used if the absolute value of the parameter is less than limit. It must be positive.
GRADIENT-CALCULATION	Determines if the gradients that have been programmed into the model functions are used (ANALYTICAL), or if one way

	(u+du) or two way (u+du and u-du) incrementation is used to determine the gradient.
U-SPACE-BOUNDS	Initialises the u-space optimisation upper bounds to Value and the u-space lower bounds to -Value.
INTERMEDIATE-RESULTS	Controls the amount of intermediate results to be generated. The possible alternatives are: NONE, LOW, MEDIUM, EXCESSIVE.
FRACTILE-FROM-PROBABILITY	Defines the optimization method used to calculate a fractile from a probability value.
UNMIN	Unconstrained minimisation in polar coordinates.
maxit	The maximal number of iterations allowed.
maxstep	The maximal number of steps in one search direction.
conv	Convergence criterion.
PROBABILITY-FROM-FRACTILE	Defines the optimization method used to calculate a probability from a fractile value.
SQP	Sequential quadratic programming.
NLPQL	Sequential quadratic programming. Extended options set. See DEFINE ... NLPQL.
RFCRC	Robusted Rackwitz-Fiessler method. See DEFINE ... RFCRC.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- PRINT ANALYSIS SETTINGS

EXAMPLE:

The following values are default when the program starts up with a new database:

```
DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION DIFFERENTIATION 1.0E-6 1.0E-3 1.0E-
6 1.0E-6 1.0E-10
DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION INTERMEDIATE-RESULTS NONE
DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION FRACTILE-FROM-PROBABILITY UNMIN 40
10 1.72633D-7
DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION PROBABILITY-FROM-FRACTILE SQP 40 10
1.72633D-7
```

DEFINE ANALYSIS-OPTION NESTED-ANALYSIS

...	NESTED-ANALYSIS	...

<table>
<tr><td rowspan="11">...</td><td rowspan="2">DIFFERENTIATION</td><td>GLOBAL</td><td rowspan="2">...</td><td rowspan="2">uspace1</td><td rowspan="2">uspace2</td><td rowspan="2">relative</td><td rowspan="2">absolute</td><td rowspan="2">limit</td></tr>
<tr><td>SYSTEM</td></tr>
<tr><td rowspan="3">GRADIENT-CALCULATION</td><td rowspan="2">GLOBAL</td><td rowspan="3">...</td><td colspan="5">ANALYTICAL</td></tr>
<tr><td colspan="5">ONEWAY-INCREMENTATION</td></tr>
<tr><td>SYSTEM</td><td colspan="5">TWOWAY-INCREMENTATION</td></tr>
<tr><td rowspan="2">U-SPACE-BOUNDS</td><td>GLOBAL</td><td rowspan="2">...</td><td colspan="5" rowspan="2">VALUE</td></tr>
<tr><td>SYSTEM</td></tr>
<tr><td rowspan="4">INTERMEDIATE-RESULTS</td><td rowspan="2">GLOBAL</td><td rowspan="4">...</td><td colspan="5">NONE</td></tr>
<tr><td colspan="5">LOW</td></tr>
<tr><td rowspan="2">SYSTEM</td><td colspan="5">MEDIUM</td></tr>
<tr><td colspan="5">EXCESSIVE</td></tr>
</table>

PURPOSE:

Define analysis options for usage of nested analyses.

PARAMETERS:

GLOBAL	Outer level of a nested analysis.
SYSTEM	Inner level of a nested analysis.
DIFFERENTIATION	Define differentiation increments for use in optimization on outer or inner level of a nested analysis.
uspace1	The differentiation increment in U-space to be used for first order derivatives. It must be positive. Used during FORM/SORM optimization.
uspace2	The differentiation increment in U-space used to be used for calculation of second order derivatives. It must be positive.
relative	Relative parameter increment. It must be positive.
absolute	Absolute parameter increment. It must be positive.
limit	Limit for application of relative parameter increment. The absolute increment is used if the absolute value of the parameter is less than limit. It must be positive.

GRADIENT-CALCULATION	Determines if the gradients that have been programmed into the model functions are used (ANALYTICAL), or if one way (u+du) or two way (u+du and u-du) incrementation is used to determine the gradient.
U-SPACE-BOUNDS	Initialises the selected level u-space optimisation upper bounds to Value and the u-space lower bounds to -Value.
INTERMEDIATE-RESULTS	Controls the amount of intermediate results to be generated on outer and inner level of a nested analysis. The possible alternatives are: NONE, LOW, MEDIUM, EXCESSIVE.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

EXAMPLE:

The following values are default when the program starts up with a new database:

```
DEFINE ANALYSIS-OPTION NESTED-ANALYSIS DIFFERENTIATION GLOBAL 1.0E-2 1.0E-2 1.0E-
3 1.0E-3 1.0E-10
DEFINE ANALYSIS-OPTION NESTED-ANALYSIS INTERMEDIATE-RESULTS NONE
```

DEFINE CONTINUOUS-PROCESS

<table>
<tr><td rowspan="3">...</td><td rowspan="3">CONTINUOUS-PROCESS</td><td>ANALYSIS-OPTION</td><td colspan="2">...</td></tr>
<tr><td>DURATION</td><td rowspan="2">...</td><td>Value</td></tr>
<tr><td>STARTING-TIME</td><td>NONE</td></tr>
</table>

PURPOSE:

Define analysis options, default duration and default starting time for a continuous stochastic process.

PARAMETERS:

DURATION	Default duration. Will be used if the continuous process does not contain a TIME variable, or if duration is not specified for the TIME variable. If the model contains no TIME variable, and a crossing rate is calculated, the duration is not used.
STARTING-POINT	Default starting point. Will be used if starting time is not specified for the TIME variable. If the model contains no TIME variable, the starting time is not used.
Value	Duration value or starting time value. Can be a numerical value or the name of a one dimensional variable.
NONE	Turn off assignment of duration value or starting point value.

NOTES:

See also:

- ASSIGN CONTINUOUS-PROCESS

EXAMPLES:

```
DEFINE CONTINUOUS-PROCESS DURATION DurVar
DEFINE CONTINUOUS-PROCESS DURATION NONE
DEFINE CONTINUOUS-PROCESS STARTING-TIME 0.0
DEFINE CONTINUOUS-PROCESS STARTING-TIME NONE
```

DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTION

<table>
<tr><td rowspan="6">...</td><td rowspan="6">ANALYSIS-OPTION</td><td rowspan="3">INTEGRATION-INTERVAL</td><td>lowerend</td></tr>
<tr><td>upperend</td></tr>
<tr><td>OFF</td></tr>
<tr><td>MINIMUM-EXTREME-VALUE</td><td>integernumber</td></tr>
<tr><td>NUMBER-OF-TIME-SPLITS</td><td>integernumber</td></tr>
<tr><td>POINTS-IN-QUADRATURE</td><td>integernumber</td></tr>
</table>

PURPOSE:

Define analysis options, default duration and default starting time for a continuous stochastic process.

PARAMETERS:

INTEGRATION-INTERVAL	Reduce integration interval for crossing rate to contributory part. The integration will be carried out between lower end and upper end.
MINIMUM-EXTREME-VALUE	The failure set is a series system of an integer number of equal but independent events.
NUMBER-OF-TIME-SPLITS	Periodicity in a stochastic process may be exploited in order to reduce the integration effort. If the number of periods (time splits) in the process is n, then the actual duration is n*D where D is the duration assigned to the process. The integration is over the assigned duration D and the calculated expected number of crossings is multiplied by n.
POINTS-IN-QUADRATURE	The number of points in the quadrature used to calculate the expected number of crossings in the duration of the process.
lowerend	Lower end of the reduced integration interval.
upperend	Upper end of the reduced integration interval.
OFF	Turn off assignment of reduced integration interval.
integernumber	Integer number.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- ASSIGN CONTINUOUS-PROCESS

- PRINT ANALYSIS-SETTINGS

EXAMPLES:

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTION INTEGRATION-INTERVAL 10 1000
```

The following values are default when the program starts up with a new database:

```
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTION INTEGRATION-INTERVAL OFF
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTION MINIMUM-EXTREME-VALUE 1
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTION NUMBER-OF-TIME-SPLITS 1
DEFINE CONTINUOUS-PROCESS ANALYSIS-OPTION POINTS-IN-QUADRATURE 6
```

DEFINE DISTRIBUTION-SIMULATION

...	DISTRIBUTION-SIMULATION	MONTE-CARLO-SIMULATION	nsim
		LATIN-HYPERCUBE-SIMULATION	nsim
		RESET	

PURPOSE:

Define analysis options for simulation of distributions.

PARAMETERS:

MONTE-CARLO-SIMULATION	Define Monte Carlo simulation of distributions.
LATIN-HYPERCUBE-SIMULATION	Define Latin Hypercube simulation of distributions.
nsim	The number of simulations to be performed.
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS
- PRINT ANALYSIS-SETTINGS
- RUN DISTRIBUTION-ANALYSIS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE DISTRIBUTION-SIMULATION MONTE-CARLO-SIMULATION 1000
DEFINE DISTRIBUTION-SIMULATION LATIN-HYPERCUBE-SIMULATION 100
```

DEFINE FORM-SORM

<table>
<tr><td rowspan="22">...</td><td rowspan="22">FORM-SORM</td><td rowspan="2">BOUNDS</td><td colspan="6">ON</td></tr>
<tr><td colspan="6">OFF</td></tr>
<tr><td colspan="2" rowspan="2">INACTIVE-CONSTRAINTS</td><td colspan="5">ON</td></tr>
<tr><td colspan="5">OFF</td></tr>
<tr><td rowspan="2">MULTINORMAL</td><td colspan="6">SQP</td></tr>
<tr><td colspan="6">CRUDE</td></tr>
<tr><td rowspan="4">NESTED-ANALYSIS</td><td rowspan="2">GLOBAL</td><td rowspan="4">...</td><td>SQP</td><td>maxit</td><td>maxstep</td><td>conv</td></tr>
<tr><td>NLPQL</td><td colspan="3">...</td></tr>
<tr><td rowspan="2">SYSTEM</td><td>RFCRC</td><td colspan="3">...</td></tr>
<tr><td>RSM</td><td colspan="3">...</td></tr>
<tr><td rowspan="4">OPTIMIZATION</td><td colspan="2">SQP</td><td>maxit</td><td>maxstep</td><td colspan="2">conv</td></tr>
<tr><td>NLPQL</td><td colspan="5">...</td></tr>
<tr><td>RFCRC</td><td colspan="5">...</td></tr>
<tr><td>RSM</td><td colspan="5">...</td></tr>
<tr><td rowspan="3">SENSITIVITY</td><td colspan="3" rowspan="2">ANALYTICAL</td><td colspan="3">ONE-WAY</td></tr>
<tr><td colspan="3">TWO-WAY</td></tr>
<tr><td colspan="6">ASYMPTOTIC</td></tr>
<tr><td rowspan="4">STARTING-POINT</td><td colspan="3" rowspan="2">INITIAL</td><td colspan="3">ASSIGNED</td></tr>
<tr><td colspan="3">DEFAULT</td></tr>
<tr><td colspan="3" rowspan="2">PARAMETER-STUDY</td><td colspan="3">PREVIOUS-SOLUTION</td></tr>
<tr><td colspan="3">SAME-AS-INITIAL</td></tr>
<tr><td>RESET</td><td colspan="6"></td></tr>
</table>

PURPOSE:

Define FORM/SORM analysis options.

PARAMETERS:

BOUNDS Control the usage of bounds in probability calculation in a large intersection. If ON, bounds are used. If OFF, the probability is calculated using the multinormal distribution on the complementary set.

INACTIVE-CONSTRAINTS	Control linearisation of constraints, that are inactive initially. If ON, such constraints are attempted linearised. If OFF, they are not linearised.
MULTINORMAL	Controls how the probability is calculated through the multinormal distribution. The SQP option is the most accurate. The CRUDE option should only be used if the SQP option fails.
NESTED-ANALYSIS	Selection of the optimization algorithm. Currently, only one algorithm is available.
GLOBAL	Outer level in a nested analysis.
SYSTEM	Inner level in a nested analysis.
OPTIMISATION	Selection of the optimization algorithm. Currently, only one algorithm is available.
SQP	Sequential quadratic programming.
maxit	The maximal number of iterations allowed.
maxstep	The maximal number of steps in one search direction.
conv	Convergence criterion.
NLPQL	Sequential quadratic programming. Extended options set. See DEFINE ... NLPQL
RFCRC	Robusted Rackwitz-Fiessler method. See DEFINE ... RFCRC.
RSM	Response surface method. See DEFINE ...RSM.
SENSITIVITY	Controls the method used to calculate parametric sensitivities and importance factors. ANALYTICAL calculation is exact for the FORM result, but requires a number of differentiations. ASYMPTOTIC calculation is quick, but not as accurate. The second order derivations using the ANALYTICAL calculation may be done ONE-WAY or TWO-WAY (to gain accuracy).
STARTING-POINT INITIAL	Controls the usage of the starting point in the FORM/SORM optimization. In a parameter study, it applies to the first analysis, as well as any other analysis where the previous solution is not used. The starting point can be either ASSIGNED (see ASSIGN STARTING-POINT) or DEFAULT. The default starting point is a small shift from the origin in U-space.
STARTING-POINT PARAMETER-STUDY	Controls the usage of starting points in a parameter study. Either the PREVIOUS-SOLUTION is used whenever possible, or the starting point is defined as above (SAME-AS-INITIAL).
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- ASSIGN STARTING-POINT
- DEFINE ANALYSIS-OPTION
- PRINT ANALYSIS-SETTINGS
- SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE FORM-SORM BOUNDS OFF
DEFINE FORM-SORM INACTIVE-CONSTRAINTS ON
DEFINE FORM-SORM MULTINORMAL SQP
DEFINE FORM-SORM NESTED-ANALYSIS GLOBAL SQP 40 10 0.0025
DEFINE FORM-SORM OPTIMIZATION SQP 40 10 0.0025
DEFINE FORM-SORM SENSITIVITY ANALYTICAL ONE-WAY
DEFINE FORM-SORM STARTING-POINT INITIAL ASSIGNED
DEFINE FORM-SORM STARTING-POINT PARAMETER-STUDY PREVIOUS-SOLUTION
```

DEFINE ... NLPQL

...	NLPQL	search method	maxit	maximum step lenght	maxfun	conv	cnsv	bestpoint

PURPOSE:

Options for NLPQL.

PARAMETERS:

search method	One of BFGS and STEEPEST-DESCENT. BFGS generates a quadratic approximation to the function optimised on. STEEPEST-DESCENT generates a sequential linear approximation and is the more robust method when the gradients have poor numerical quality.
maxit	Maximum number of general iterations (gradient evaluations.)
maximum step length	FREE (limited by optimisation bounds) or VALUE. The value is the maximum steplength during one iteration. Prevents overshooting.
maxfun	Maximum number of function evaluations in line search for step length that improves merit function.
conv	Kuhn-Tucker optimality criterion.
cnsv	Test for constraint violation. ON-DEFAULT uses the square root of conv as test value. ON-USER uses a user specified value as test value. OFF skips the constraint value test.
bestpoint	ON delivers the best point reached during optimisation even if a convergency criterion is not met. OFF delivers a point that necessarily fulfils the convergency criteria.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- PRINT ANALYSIS-SETTINGS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE FORM-SORM OPTIMIZATION NLPQL BFGS 40 VALUE 5.0 10 0.0001726 ON-DEFAULT OFF
```

DEFINE ... RFCRC

...	RFCRC	method	maxit	conv	test

PURPOSE:

Options for RFCRC.

PARAMETERS:

method	One of RF (Racwitz-Fiessler method) and RFCRC (Rackwitz-Fiessler method robusted with circle steps.)
maxit	Maximum number of general iterations (gradient evaluations.)
conv	Optimality criterion. Test for the U-space distance between the two last iterates.
test	Progress test. If RFstep(i+1) suggested by the algorithm is less than RFstep(i)/test, then accept the step, else proceed with a circle step.

NOTES:

A RF step is performed initially. Then the next step suggested by the RF method is examined. If the progress is unsatisfactory, then a circle step is performed. This step defines a u-space circle with center at u=0 and passing through the current iteration point in the plane defined by the u-space gradient at that point. The minimum point, um, of the event function, g(u), on this circle is found and an iteration is performed on the line from 0 to um to find g(unext)=0.

The method is restricted to a single event.

The analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- PRINT ANALYSIS-SETTINGS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE FORM-SORM OPTIMIZATION RFCRC CIRCLE 40 0.001 4.0
```

DEFINE ... RSM

...	RSM	method	contribution	maxit	conv	range	incrementation	initial increment	...

...	reduction factor	minimum increment	zero limit

PURPOSE:

Options for RSM.

PARAMETERS:

method	One of LINEAR, LINQUADIAG and LINQUA. LINEAR approximates the function with a linear surface. LINQUADIAG includes the diagonal terms of a quadratic approximation. LINQUA estimates a full quadratic approximation.
contribution	LASTPOINT means that only the last point is used for function approximation. RADIUSCONTR means that experiments generated around a point that is closer than range to the current iteration point contributes to the response surface generation. It contributes only if the incremnt used at that point is also less than range.
maxit	Maximum number of general iterations (response surface approximations.)
conv	Optimality criterion. When the u-space distance between two successive iterates is less than conv, the iteration stops.
range	As described above (contribution).
incrementation	ONEWAY or TWOWAY. One way means that an experiment is defined at u+delta. TWOWAY means that an experiment is also defined at u-delta.
initial increment	delta at the starting point for the iterations.
reduction factor	delta is divided by the reduction factor once for each new iterate.
minimum increment	The minimum delta. If the reduction yields a value less than this value, the minimum value is used.
zero limit	The normalised gradient a of g(u) at u=0 is approximated from the first response surface. The smalles values of ai2 are summed up, S, until the zero limit is reached (the next contribution violates the limit). The corresponding ui variables are kept constant at zero value during the iterations. The final reliability index is multiplied by the omission factor 1/(1-S)1/2.

NOTES:

The response surface method fits a linear or a (partly) quadratic function to a set of points (experiments). The "design point" is found for the zero surface implied by the response function. A new set of experiments is generated around this point. The new information, and possibly previous experiments, are used to generate a new resonse function. This is repeated until convergency.

The method is restricted to a single event.

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- PRINT ANALYSIS-SETTINGS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE FORM-SORM OPTIMIZATION RSM LINQUADIAG RADIUSCONTR 40 0.01 0.5 TWOWAY 0.1
3.0 0.1 0.01
```

DEFINE MEAN-VALUE-FORM

<table>
<tr><td rowspan="6">...</td><td rowspan="6">MEAN-VALUE-FORM</td><td>POINTS</td><td>number</td></tr>
<tr><td>LOWER-PROBABILITY</td><td>lower</td></tr>
<tr><td>UPPER-PROBABILITY</td><td>upper</td></tr>
<tr><td rowspan="2">GRADIENT</td><td>ONE</td></tr>
<tr><td>THREE</td></tr>
<tr><td colspan="2">RESET</td></tr>
</table>

PURPOSE:

Define Mean value based FORM analysis options.

PARAMETERS:

POINTS number	The number of points to be calculated. These are spaced equally in distance in U-space, from the distance corresponding to lower probability bound to the distance corresponding to upper probability bound.
LOWER-PROBABILITY lower	The lower probability bound for the range in which values are calculated. Must be positive and less than 1.
UPPER-PROBABILITY upper	The upper probability bound for the range in which values are calculated. Must be positive and less than 1.
GRADIENT	The method uses either ONE gradient (at the origin of U-space) or THREE gradients (the remaining two are calculated at the lower and upper bound).
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

See also:

- PRINT ANALYSIS-SETTINGS
- SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE MEAN-VALUE-FORM POINTS 19
DEFINE MEAN-VALUE-FORM LOWER-PROBABILITY 0.01
DEFINE MEAN-VALUE-FORM UPPER-PROBABILITY 0.99
DEFINE MEAN-VALUE-FORM GRADIENT ONE
```

DEFINE PARAMETER-STUDY

...	PARAMETER-STUDY	parameter	{	value	}*

PURPOSE:

Define parameter study values of a fixed variable or of a numerical parameter in a distribution or of a numerical argument in a function.

PARAMETERS:

parameter	The name of a fixed variable or the name of a numerical parameter in a distribution or of a numerical argument in a function.
value	Those parameter values for which the parameter study is to be performed.

NOTES:

1 A parameter study may be modified by entering the command again and selecting the same parameter. The current values are then presented as defaults.

2 Usage of the parameter study is controlled by the command DEFINE ANALYSIS-OPTION PARAMETER-STUDY.

3 This command is described in the User's Manual for Proban Version 3 as ASSIGN PARAMETER-STUDY.

See also:

- DEFINE ANALYSIS-OPTION PARAMETER-STUDY
- PRINT PARAMETER-STUDY
- PRINT RESULT PARAMETER-STUDY
- DISPLAY RESULT PARAMETER-STUDY

EXAMPLES:

```
DEFINE PARAMETER-STUDY StrCorr GROUP 0.1 0.9 0.1
DEFINE PARAMETER-STUDY XX-abc ( ONLY 22 24 25 29.6 )
```

DEFINE PRESENTATION

...	PRESENTATION	FUNCTION	...
		RESULT	...

PURPOSE:

Define presentation of results and input data.

PARAMETERS:

FUNCTION	Define presentation of model functions.
RESULT	Define presentation of analysis results.

NOTES:

None.

DEFINE PRESENTATION FUNCTION

...	FUNCTION	1D-FUNCTION-DISPLAY	nval	
		2D-FUNCTION-DISPLAY	nx	ny

PURPOSE:

Define options for presentation of model functions.

PARAMETERS:

1D-FUNCTION-DISPLAY nval	The number of function evaluations used in a one dimensional graph of a model function.
2D-FUNCTION-DISPLAY nx ny	The number of abscissa (nx) and ordinate (ny) values used in a two dimensional display of a model function. The total number of function evaluations will be nx*ny.

NOTES:

See also:

- DISPLAY FUNCTION

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE PRESENTATION FUNCTION 1D-FUNCTION-DISPLAY 101
DEFINE PRESENTATION FUNCTION 2D-FUNCTION-DISPLAY 21 21
```

DEFINE PRESENTATION RESULT

...	RESULT	CONFIDENCE-VALUE	conf	
		IMPORTANCE-CUTOFF	cutoff	
		IMPORTANCE-LIMIT	limit	
		INTERMEDIATE-SIMULATIONS	intsim	
		SENSITIVITY-MEASURE	inc	lim
		V-SPACE-POINT	ON	
			OFF	
		RESET		

PURPOSE:

Define options for presentation of results.

PARAMETERS:

CONFIDENCE-VALUE conf
: The confidence value that is used with print and display of confidence limits. This value must be given in %, e.g. a value of 95 will print/display 95% confidence limits.

IMPORTANCE-CUTOFF cutoff
: This value is used to cut off the smallest importance factor values from the print of importance factors. This value must be given in %, e.g. if input as 5, all importance factor values less than 5% will not be printed.

IMPORTANCE-LIMIT limit
: This value is used to group the smallest importance factor values in the display of importance factors. This value must be given in %, e.g. if input as 5, all importance factor values less than 5% will be shown in one pie slice, named “Other”.

INTERMEDIATE-SIMULATIONS intsim
: Determines how many lines of intermediate results will be printed with the PRINT RESULT ALL command after a simulation analysis. To see all intermediate simulation results, set (intsim) to a value equal to or greater than the number of simulations performed.

SENSITIVITY-MEASURE inc lim
: Defines how sensitivity measures are calculated. A sensitivity measure is dimensionless, in that it measures the change in the target value when a parameter is multiplied by (1+(inc)). As this definition does not work when the parameter value is zero, (lim) denotes the smallest parameter value to which it can be applied.

V-SPACE-POINT	Defines if the V-space coordinates of a FORM linearisation point are to be printed (ON) or not (OFF).
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

This command is documented in the Proban Users Manual as DEFINE RESULT-OPTION.

See also:

- PRINT RESULT
- DISPLAY RESULT
- CHANGE TRANSFER-FUNCTION
- CREATE TRANSFER-FUNCTION
- DISPLAY TRANSFER-FUNCTION
- PRINT TRANSFER-FUNCTION

EXAMPLE:

The following values are default when the program starts up with a new database:

```
DEFINE PRESENTATION RESULT CONFIDENCE-VALUE 90
DEFINE PRESENTATION RESULT IMPORTANCE-CUTOFF 0
DEFINE PRESENTATION RESULT IMPORTANCE-LIMIT 5
DEFINE PRESENTATION RESULT INTERMEDIATE-SIMULATIONS 10
DEFINE PRESENTATION RESULT SENSITIVITY-MEASURE 0.1 0.0001
DEFINE PRESENTATION RESULT V-SPACE-POINT OFF
```

DEFINE PROBABILITY-SIMULATION

...	PROBABILITY-ANALYSIS	AXIS-ORTHOGONAL	...
		DESIGN-POINT	...
		DIRECTIONAL	...
		MONTE-CARLO	...

PURPOSE:

Define analysis options that apply to simulation of a probability.

PARAMETERS:

AXIS-ORTHOGONAL	Define analysis options for axis orthogonal simulation.
DESIGN-POINT	Define analysis options for design point simulation.
DIRECTIONAL	Define analysis options for directional simulation.
MONTE-CARLO	Define analysis options for Monte Carlo simulation.

NOTES:

None

DEFINE PROBABILITY-SIMULATION AXIS-ORTHOGONAL

<table>
<tr><td rowspan="9">...</td><td rowspan="9">AXIS-ORTHOGONAL</td><td>COEFFICIENT-OF-VARIATION</td><td>cov</td></tr>
<tr><td>CPU-TIME</td><td>cpu</td></tr>
<tr><td rowspan="2">DENSITY</td><td>CONDITIONED</td></tr>
<tr><td>STANDARD-NORMAL</td></tr>
<tr><td rowspan="3">SEARCH</td><td>RISKY-AND-FAST</td></tr>
<tr><td>MEDIUM-SAFE</td></tr>
<tr><td>SAFE-AND-SLOW</td></tr>
<tr><td>SIMULATIONS</td><td>nsim</td></tr>
<tr><td colspan="2">RESET</td></tr>
</table>

PURPOSE:

Define analysis options for axis orthogonal simulation of a probability.

PARAMETERS:

COEFFICIENT-OF-VARIATION cov	The simulations will stop if the coefficient of variation of the simulated result becomes lower than or equal to cov. To disable this stop criterion, set cov to 0. cov must be non-negative.
CPU-TIME cpu	The simulations will stop when the cpu time cpu (in seconds) has been exceeded. The check is performed after each simulation is completed. To disable this stop criterion, set cpu to 0. cpu must be non-negative.
DENSITY	Specifies the sampling density.
CONDITIONED	This density has a shape that is dependent on the shape of the limit state surface, and produces a result that is a multiplicative correction to the FORM probability. This is generally quite fast and accurate, but it depends on a reasonable FORM approximation to the limit state surface.
STANDARD-NORMAL	This density is not dependent on the shape of the limit state surface, and produces an additive correction to the FORM probability. This option is the slowest and safest of the two.
SEARCH	Specifies how the line search for points on the limit state surface is performed along the simulated direction.
RISKY-AND-FAST	This search method simply checks one point far out on the line, and looks for a solution only if the sign of the function is different at the origin and at the end point. This method is generally

	sufficient for single events. It is generally not recommended for analysis of other events.
MEDIUM-SAFE	This search method steps out to the first solution (if any), then takes one step to the end to see if there should be another solution. This method is sufficiently accurate in most cases.
SAFE-AND-SLOW	This search method steps out to the "end" of the line (where the probability becomes negligible) without skipping any larger pieces.
SIMULATIONS nsim	The simulation will stop after nsim simulations has been completed. nsim must be a positive whole number.
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

1 The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

2 The simulation will run until any one of the stop criteria has been met.

3 Sensitivity calculation is not possible with this analysis method.

See also:

- PRINT ANALYSIS-SETTINGS
- SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS3

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE PROBABILITY-ANALYSIS AXIS-ORTHOGONAL COEFFICIENT-OF-VARIATION 0
DEFINE PROBABILITY-ANALYSIS AXIS-ORTHOGONAL CPU-TIME 60
DEFINE PROBABILITY-ANALYSIS AXIS-ORTHOGONAL DENSITY CONDITIONED
DEFINE PROBABILITY-ANALYSIS AXIS-ORTHOGONAL SEARCH MEDIUM-SAFE
DEFINE PROBABILITY-ANALYSIS AXIS-ORTHOGONAL SIMULATIONS 50
```

DEFINE PROBABILITY-SIMULATION DESIGN-POINT

...	DESIGN-POINT	COEFFICIENT-OF-VARIATION	cov
		CPU-TIME	cpu
		SIMULATIONS	nsim
		RESET	

PURPOSE:

Define analysis options for design point simulation of a probability.

PARAMETERS:

COEFFICIENT-OF-VARIATION cov	The simulations will stop if the coefficient of variation of the simulated result becomes lower than or equal to cov. To disable this stop criterion, set cov to 0. cov must be non-negative.
CPU-TIME cpu	The simulations will stop when the cpu time cpu (in seconds) has been exceeded. The check is performed after each simulation is completed. To disable this stop criterion, set cpu to 0. cpu must be non-negative.
SIMULATIONS nsim	The simulation will stop after nsim simulations has been completed. nsim must be a positive whole number.
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

1 The design point simulation first finds the design point. Then it performs a Monte Carlo probability simulation with sampling density centered at the design point.

2 The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

3 The simulation will run until any one of the stop criteria has been met.

4 Sensitivity calculation is not possible with this analysis method.

See also:

- PRINT ANALYSIS-SETTINGS
- SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE PROBABILITY-ANALYSIS DESIGN-POINT COEFFICIENT-OF-VARIATION 0
DEFINE PROBABILITY-ANALYSIS DESIGN-POINT CPU-TIME 60
DEFINE PROBABILITY-ANALYSIS DESIGN-POINT SIMULATIONS 1000
```

DEFINE PROBABILITY-SIMULATION DIRECTIONAL

<table>
<tr><td rowspan="15">...</td><td rowspan="15">DIRECTIONAL</td><td>COEFFICIENT-OF-VARIATION</td><td colspan="2">cov</td></tr>
<tr><td>CPU-TIME</td><td colspan="2">cpu</td></tr>
<tr><td rowspan="5">METHOD</td><td colspan="2">DEFAULT</td></tr>
<tr><td colspan="2">RANDOM-DIRECTION</td></tr>
<tr><td colspan="2">ORTHOGONAL-1</td></tr>
<tr><td colspan="2">ORTHOGONAL-2</td></tr>
<tr><td colspan="2">ORTHOGONAL-3</td></tr>
<tr><td rowspan="3">SEARCH</td><td colspan="2">RISKY-AND-FAST</td></tr>
<tr><td colspan="2">MEDIUM-SAFE</td></tr>
<tr><td colspan="2">SAFE-AND-SLOW</td></tr>
<tr><td rowspan="2">SEARCH-LIMIT</td><td>PROBABILITY</td><td>probvalue</td></tr>
<tr><td>STANDARD-NORMAL</td><td>argvalue</td></tr>
<tr><td>STEP-LENGTH</td><td colspan="2">length</td></tr>
<tr><td>SIMULATIONS</td><td colspan="2">nsim</td></tr>
<tr><td colspan="3">RESET</td></tr>
</table>

PURPOSE:

Define analysis options for directional simulation of a probability.

PARAMETERS:

COEFFICIENT-OF-VARIATION *cov*	The simulations will stop if the coefficient of variation of the simulated result becomes lower than or equal to *cov.* To disable this stop criterion, set *cov* to 0. The *cov* must be non-negative.
CPU-TIME *cpu*	The simulations will stop when the cpu time (in seconds) has been exceeded. The check is performed after each simulation is completed. To disable this stop criterion, set *cpu* to 0. The *cpu* must be non-negative.
METHOD	Specifies the sampling method.
DEFAULT	The default sampling method is selected on the basis of the dimension of the *u*-space. This method is recommended in most cases. If the model contains a time consuming model function, it may be better to use the random direction method.

RANDOM-DIRECTION	The probability is calculated in a simulated direction and in the opposite direction, and the average of the two probabilities is used as the sample probability. This reduces the sample variance because the two probabilities can be assumed to be negatively correlated. This is the simplest technique. It is mostly useful when the more sophisticated techniques take too long time to produce results.
ORTHOGONAL-1	An orthogonal set of directions, that span the *u*-space, is simulated. The probability is then found in each of these directions and their opposite directions, and the average value is calculated and used as the sample value. The sample variance is further reduced by this method. The drawback is that it may take some time to produce each sample value because of the large number of calculations involved.
ORTHOGONAL-2	Is a sophistication of the ORTHOGONAL-1 method. Instead of using the simulated directions and their opposites, all possible averages of two of these directions are used. This gives a better coverage of *u*-space, but increases computation time considerably.
ORTHOGONAL-3	As ORTHOGONAL-2, except that averages are formed of all possible combinations of three directions instead of two. This method can be very time consuming.
SEARCH	Specifies how the line search for points on the limit state surface is performed along the simulated direction.
RISKY-AND-FAST	This search method simply checks one point far out on the line, and looks for a solution only if the sign of the function is different at the origin and at the end point. This method is generally sufficient for single events. It is generally not recommended for analysis of other events.
MEDIUM-SAFE	This search method steps out to the first solution (if any), then takes one step to the end to see if there should be another solution. This method is sufficiently accurate in most cases.
SAFE-AND-SLOW	This search method steps out in the *u*-space to the “end” of the line (where the probability becomes negligible) without skipping any larger pieces.
SEARCH-LIMIT	The search method steps out in the *u*-space until the probability of the remaining line becomes negligible, as specified by the search limit. The search limit may be entered as a PROBABILITY with value *probval* or as a STANDARD-NORMAL *argval*, which is the *u*-space search limit. Notice the correspondence ($\Phi(-u)$=*probval*)
STEP-LENGTH *length*	The search method steps out in the *u*-space in search for zero points until the probability of the remaining line becomes

	negligible, as specified by the search limit. Starting from u=0, the next step is $u_{next} = u_{current} + length$.
SIMULATIONS *nsim*	The simulation will stop after *nsim* simulations has been completed. nsim must be a positive whole number.
RESET	Reset all values and options to the default values used when initialising a new database.

NOTES:

1 The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

2 The simulation will run until any one of the stop criteria has been met.

See also:

- PRINT ANALYSIS-SETTINGS
- SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE PROBABILITY-ANALYSIS DIRECTIONAL COEFFICIENT-OF-VARIATION 0
DEFINE PROBABILITY-ANALYSIS DIRECTIONAL CPU-TIME 60
DEFINE PROBABILITY-ANALYSIS DIRECTIONAL METHOD DEFAULT
DEFINE PROBABILITY-ANALYSIS DIRECTIONAL SEARCH MEDIUM-SAFE
DEFINE PROBABILITY-ANALYSIS DIRECTIONAL SIMULATIONS 50
```

DEFINE PROBABILITY-SIMULATION MONTE-CARLO

...	MONTE-CARLO	COEFFICIENT-OF-VARIATION	cov
		CPU-TIME	cpu
		SIMULATIONS	nsim
		RESET	

PURPOSE:

Define analysis options for Monte Carlo simulation of a probability.

PARAMETERS:

COEFFICIENT-OF-VARIATION cov — The simulations will stop if the coefficient of variation of the simulated result becomes lower than or equal to <cov>. To disable this stop criterion, set cov to 0. cov must be non-negative.

CPU-TIME cpu — The simulations will stop when the cpu time cpu (in seconds) has been exceeded. The check is performed after each simulation is completed. To disable this stop criterion, set cpu to 0. cpu must be non-negative.

SIMULATIONS nsim — The simulation will stop after nsim simulations has been completed. nsim must be a positive whole number.

RESET — Reset all values and options to the default values used when initialising a new database.

NOTES:

1 The current analysis settings may be printed by use of the PRINT ANALYSIS-SETTINGS command.

2 The simulation will run until any one of the stop criteria has been met.

3 Sensitivity calculation is not possible with this analysis method.

See also:

- PRINT ANALYSIS-SETTINGS
- SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS

EXAMPLES:

The following values are default when the program starts up with a new database:

```
DEFINE PROBABILITY-ANALYSIS MONTE-CARLO COEFFICIENT-OF-VARIATION 0
DEFINE PROBABILITY-ANALYSIS MONTE-CARLO CPU-TIME 60
```

```
DEFINE PROBABILITY-ANALYSIS MONTE-CARLO SIMULATIONS 1000
```

DEFINE TWO-PARAMETER-STUDY

...	TWO-PARAMETER-STUDY	parameter1	{	value1	}*	parameter2	{	value2	}*

PURPOSE:

Define two-parameter study values. Each parameter is one of a fixed variable or of a numerical parameter in a distribution or of a numerical argument in a function.

PARAMETERS:

parameter1	The name of a fixed variable or the name of a numerical parameter in a distribution or of a numerical argument in a function.
value1	Those parameter values of parameter1 for which the parameter study is to be performed.
parameter2	The name of a fixed variable or the name of a numerical parameter in a distribution or of a numerical argument in a function.
value2	Those parameter values of parameter2 for which the parameter study is to be performed.

NOTES:

1 The parameter study is performed over the matrix (value1,value2) so that all combinations of values are covered.

2 A parameter study may be modified by entering the command again and selecting the same parameters. The current values are then presented as defaults.

3 Usage of the parameter study is controlled by the command DEFINE ANALYSIS-OPTION PARAMETER-STUDY.

4 This command is described in the User's Manual for Proban Version 3 as ASSIGN PARAMETER-STUDY.

See also:

- DEFINE ANALYSIS-OPTION PARAMETER-STUDY
- PRINT PARAMETER-STUDY
- PRINT RESULT PARAMETER-STUDY
- DISPLAY RESULT PARAMETER-STUDY

EXAMPLES:

```
DEFINE TWO-PARAMETER-STUDY StrCorr GROUP 0.1 0.9 0.1 StrStd GROUP 1 2 0.2
```

```
DEFINE TWO-PARAMETER-STUDY XX-abc ( ONLY 22 24 25 29.6 ) XX-def ( ONLY 3 4 5 )
```

DELETE

DELETE	EVENT	...
	FUNCTION	...
	RESULT	...
	VARIABLE	...

PURPOSE:

Delete a named object.

PARAMETERS:

EVENT	Delete an event.
FUNCTION	Delete a function formula.
RESULT	Delete an analysis result.
VARIABLE	Delete a random variable.

NOTES:

None.

DELETE EVENT

...	EVENT	name+

PURPOSE:

Delete one or more events.

PARAMETERS:

name+	Name(s) of the event(s) to be deleted.

NOTES:

Deletion cannot be undone. The only way to undo a deletion is to edit the command(s) generating the deleted object from the journal file, and then read the command input file into the program again.

See also:

- CREATE EVENT
- CHANGE EVENT
- COPY EVENT
- RENAME EVENT
- DISPLAY EVENT
- PRINT EVENT

EXAMPLES:

```
DELETE EVENT PFC*
```

DELETE FUNCTION

...	FUNCTION	name+

PURPOSE:

Delete one or more function formulas or function integrals.

PARAMETERS:

name+	Name(s) of the function formula(s) to be deleted.

NOTES:

Deletion cannot be undone. The only way to undo a deletion is to edit the command(s) generating the deleted object from the journal file, and then read the command input file into the program again.

See also:

- CREATE FUNCTION
- CHANGE FUNCTION
- RENAME FUNCTION
- DISPLAY FUNCTION
- PRINT FUNCTION

EXAMPLES:

```
DELETE FUNCTION SYMFUN
```

DELETE RESULT

...	RESULT	name+

PURPOSE:

Delete one or more results.

PARAMETERS:

name+	Name(s) of the result(s) to be deleted.

NOTES:

1 Deletion cannot be undone. The only way to undo a deletion is to edit the command(s) generating the deleted object from the journal file, and then read the command input file into the program again.

2 Those results created by this program should not be deleted by the user.

See also:

- RUN PROBABILITY-ANALYSIS
- RUN DISTRIBUTION-ANALYSIS
- SAVE RESULT
- DISPLAY RESULT
- PRINT RESULT

EXAMPLES:

```
DELETE RESULT Prob*
```

DELETE VARIABLE

...	VARIABLE	name+

PURPOSE:

Delete one or more variables.

PARAMETERS:

name+	Name(s) of the variable(s) to be deleted.

NOTES:

1 Deletion cannot be undone. The only way to undo a deletion is to edit the command(s) generating the deleted object from the journal file, and then read the command input file into the program again.

2 If a deleted variable is used in a single event, the single event is also deleted.

See also:

- CREATE VARIABLE
- CHANGE VARIABLE
- COPY VARIABLE
- RENAME VARIABLE
- DISPLAY VARIABLE
- PRINT VARIABLE

EXAMPLES:

```
DELETE VARIABLE X*
```

DISPLAY

DISPLAY	DISTRIBUTION	...
	EVENT	...
	FUNCTION	...
	FITTED-DISTRIBUTION	...
	RESULT	...

PURPOSE:

To present input data and results graphically.

PARAMETERS:

DISTRIBUTION	Display the distribution of random variable(s).
EVENT	Display an event.
FUNCTION	Display a model function.
FITTED-DISTRIBUTION	Display a fitted the distribution with input data.
RESULT	Display an analysis result.

NOTES:

Display of results will only be available when the results exist.

DISPLAY DISTRIBUTION

...	DISTRIBUTION	univar+	DENSITY
			DISTRIBUTION
			COMPLEMENTARY-DISTRIBUTION

PURPOSE:

Display distribution and density functions for existing variables.

PARAMETERS:

univar+	A selection of one-dimensional distribution variables with numerical or fixed parameters.
DENSITY	Display the density function for the selected variable(s).
DISTRIBUTION	Display the distribution function for the selected variable(s).
COMPLEMENTARY-DISTRIBUTION	Display the complementary distribution function for the selected variable(s).

NOTES:

The functions are calculated within a range of three standard deviations (five standard deviations if limited by a bound) on each side of the mean.

See also:

- DISPLAY RESULT DISTRIBUTION
- DISPLAY FITTED-DISTRIBUTION
- PRINT DISTRIBUTION
- PRINT VARIABLE
- SET GRAPH

EXAMPLES:

```
DISPLAY DISTRIBUTION ( ONLY Width Height ) DENSITY
```

DISPLAY EVENT

...	EVENT	event	SINGLE
			MULTIPLE

PURPOSE:

Display the definition of an event as a network.

PARAMETERS:

event	The name of the event to be displayed.
SINGLE	Display only the first level subevents.
MULTIPLE	Display the first two levels of subevents as network.

NOTES:

Unions are displayed horizontally and intersections vertically.

See also:

- PRINT EVENT
- SET

EXAMPLES:

```
DISPLAY EVENT Beam-Fail MULTIPLE
```

DISPLAY FITTED-DISTRIBUTION

...	FITTED-DISTRIBUTION	variable

PURPOSE:

Display a fitted distribution with the points it is fitted to.

PARAMETERS:

variable	Name of a variable assigned a fitted distribution.

NOTES:

1 A spline fit or cumulative fit is displayed as a distribution function curve.

2 A fit to observations is displayed as a histogram with the density function of the fitted distribution. This display can be regulated by use of the SET GRAPH HISTOGRAM command.

See also:

- DISPLAY DISTRIBUTION
- PRINT DISTRIBUTION
- PRINT VARIABLE
- SET

EXAMPLES:

```
DISPLAY FITTED-DISTRIBUTION Spl33
```

DISPLAY FUNCTION

...	FUNCTION	name	[coord]	from	...

<table>
<tr><td rowspan="3">...</td><td>ONE-ARGUMENT</td><td>argx</td><td colspan="7">tox</td></tr>
<tr><td rowspan="2">TWO-ARGUMENTS</td><td rowspan="2">argx</td><td rowspan="2">tox</td><td rowspan="2">argy</td><td rowspan="2">toy</td><td colspan="4">SURFACE</td></tr>
<tr><td>CONTOUR</td><td>min</td><td>max</td><td>step</td></tr>
</table>

PURPOSE:

Display distribution and density functions for existing variables.

PARAMETERS:

name	Name of the function.
[coord]	Coordinate of the function, if multidimensional.
from	Argument value(s) where the calculation of the function is started.
ONE-ARGUMENT	Display the function as a graph with one argument along the abscissa and the function value as the ordinate.
argx	Name of the argument to be used as abscissa.
tox	End value along the abscissa axis.
TWO-ARGUMENT	Display the function as a surface or contour plot. This option is not available for functions with only one argument.
argy	Name of the argument to be used as ordinate.
toy	End value along the ordinate axis.
SURFACE	Show a surface plot.
CONTOUR	Show a contour plot.
min,max,step	Contour specification: min, min+step, ... until max is reached.

NOTES:

1 Functions where the number of coordinates is defined by the user cannot be displayed.

2 The function option values in effect at the time of display will be used. Note that these may affect the number of arguments of the function, as well as the dimension and function value.

See also:

- DEFINE PRESENTATION FUNCTION
- PRINT FUNCTION
- ASSIGN FUNCTION-OPTION
- SET GRAPH

EXAMPLES:

```
DISPLAY FUNCTION Power 0.0 2.34 ONE-ARG Value 7.0
DISPLAY FUNCTION Power 0.0 1.0 TWO-ARG Value 5.0 Exponent 3.0 SURFACE
```

DISPLAY RESULT

...	RESULT	DISTRIBUTION	...
		IMPORTANCE-FACTORS	...
		PARAMETER-STUDY	...

PURPOSE:

Display results generated by Proban graphically.

PARAMETERS:

DISTRIBUTION	Display the result of a distribution analysis.
IMPORTANCE-FACTORS	Display the importance factors resulting from a probability analysis.
PARAMETER-STUDY	Display results as a function of the parameters in a parameter study.

NOTES:

None.

DISPLAY RESULT DISTRIBUTION

<table>
<tr><td rowspan="3">...</td><td rowspan="3">DISTRIBUTION</td><td rowspan="3">[value1]</td><td rowspan="3">[value2]</td><td rowspan="3">univar/coordinate/result+</td><td>DENSITY</td></tr>
<tr><td>DISTRIBUTION</td></tr>
<tr><td>COMPLEMENTARY-DISTRIBUTION</td></tr>
</table>

PURPOSE:

Display distribution and density functions for existing variables and for results.

PARAMETERS:

[value1]	This input is only required if the selected result is a parameter study. Value1 is then one of the first-parameter values for which the study was run. The particular result from the analysis using the selected value will be displayed.
[value2]	This input is only required if the selected result is a two-parameter study. Value2 is then one of the second-parameter values for which the study was run. The particular result from the analysis using the selected value will be displayed.
univar/coordinate/result+	A selection of one-dimensional distribution variables with numerical or fixed parameters, or of results. The following results may be available: Empirical: The empirical distribution from a simulation. Mean-V-FORM: The distribution calculated in a Mean value based FORM analysis.
DENSITY	Display the density function for the selected variable(s). For an empirical distribution, a histogram is drawn (see also SET GRAPH HISTOGRAM). It is not possible to display the density for a Mean-V-FORM result.
DISTRIBUTION	Display the distribution function for the selected variable(s).
COMPLEMENTARY-DISTRIBUTION	Display the complementary distribution function for the selected variable(s).

NOTES:

1 The distribution and density functions are calculated within a range of three standard deviations on each side of the mean.

2 When a distribution simulation is selected, and no parameter study was performed, two variables are fitted to the estimated moments: a Hermite transformation distribution (using four moments) and a Normal distribution (using two moments). These are available in the variables named Hermite-Fit and Normal-Fit.

3 A histogram cannot be displayed with a logarithmic X or Y axis.

4 The empirical distribution function is calculated as: F($x_{(i)}$) = i / (n+1) when n simulations were completed and the sample points have been ordered as: $x_{(1)}$ £ $x_{(2)}$ £ ... £ $x_{(n)}$.

See also:

- DISPLAY DISTRIBUTION
- PRINT RESULT
- SELECT RESULT
- SET

EXAMPLE:

```
DISPLAY RESULT DISTRIBUTION ( ONLY Empirical Normal-Fit ) DENSITY
DISPLAY RESULT DISTRIBUTION ONLY Mean-V-FORM DISTRIBUTION
```

DISPLAY RESULT IMPORTANCE-FACTORS

...	IMPORTANCE-FACTORS	[value1+]	[value2+]

PURPOSE:

Display importance factors.

PARAMETERS:

value1+	This input is only required if the selected result is a parameter study. value1 is then a selection of the first-parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be displayed.
value2+	This input is only required if the selected result is a two parameter study. Value2 is then a selection of the second-parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be displayed.

NOTES:

1 The importance factors are displayed as a pie chart.

2 All importance factor values less than a user definable limit are grouped into one segment labelled “Other” (see DEFINE PRESENTATION RESULT IMPORTANCE-LIMIT).

3 The form of the pie charts may be manipulated by use of the command SET GRAPH PIE-CHART.

4 Examples of the display can be seen in Figure 3.4 and Figure 3.14.

See also:

- DEFINE PRESENTATION RESULT IMPORTANCE-LIMIT
- DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR
- PRINT RESULT
- SELECT RESULT
- SET

EXAMPLES:

```
DISPLAY RESULT IMPORTANCE-FACTORS              % no parameter study
DISPLAY RESULT IMPORTANCE-FACTORS ONLY 22.5    % pick one from a study
DISPLAY RESULT IMPORTANCE-FACTORS ONLY *       % all results from a study
```

DISPLAY RESULT PARAMETER-STUDY

...	PARAMETER-STUDY	IMPORTANCE-FACTOR	...
		MAIN-RESUL	...

PURPOSE:

Display results as a function of the parameters in a parameter study.

PARAMETERS:

IMPORTANCE-FACTOR	Display importance factors as a function of the parameters.
MAIN-RESULT	Display one or more main results as a function of the parameters.

NOTES:

None.

DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR

...	IMPORTANCE-FACTOR	variable

PURPOSE:

Display importance factor for variable as a function of the parameters in a parameter study.

PARAMETERS:

variable Name of variable or importance group.

NOTES:

An example of the display can be seen in Figure 3.14.

See also:

- PRINT RESULT PARAMETER-STUDY IMPORTANCE-FACTOR
- DISPLAY RESULT IMPORTANCE-FACTORS
- SELECT RESULT
- SET

EXAMPLES:

```
DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR ( ONLY Depth ImpGroup-1 )
DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR ONLY T*
```

DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT

...	MAIN-RESULT	mainres+	[coordinate]

PURPOSE:

Display main results as a function of the parameters in a parameter study.

PARAMETERS:

mainres+	A selection of main results. The list of available results depend on the analysis performed. All possible main results are presented in the list, even though they may not all be calculated for all the individual analyses in the parameter study. For deterministic analysis of a variable there will be one result for each coordinate, and for an event there will be one result. These results will be named after the variable or event analysed.
coordinate+	A coordinate of a vector if a vector variable with more than one coordinate is sampled.

NOTES:

See also:

- PRINT RESULT PARAMETER-STUDY MAIN-RESULT
- SELECT RESULT
- SET

EXAMPLES:

```
DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT ( ONLY Prob* Conf* )
DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT ONLY *Mean*
```

EXIT

EXIT

PURPOSE:

Close all open files and stop execution of Proban

PARAMETERS:

None.

NOTES:

1 This command is not available from the menu bar in graphics mode. Use FILE EXIT instead.

2 This command is not journalled.

3 EXIT cannot be abbreviated.

EXAMPLES:

```
EXIT
```

FILE

FILE	PLOT
	EXIT

PURPOSE:

To manage file access and close the program.

PARAMETERS:

EXIT	Close all open files and exit the program. See the command description for EXIT.
PLOT	Execute the last DISPLAY command and write the result to the currently selected plot file. See the command description for PLOT.

NOTES:

None.

GET

<table>
<tr><td rowspan="7">GET</td><td>U-SPACE</td><td>event</td><td>{</td><td>variable</td><td>xvalue</td><td colspan="2">}*</td></tr>
<tr><td>X-SPACE</td><td>event</td><td>{</td><td>variable</td><td>uvalue</td><td colspan="2">}*</td></tr>
<tr><td rowspan="5">RESULT</td><td>MAIN-RESULT</td><td colspan="5">mresname</td></tr>
<tr><td>SENSITIVITY</td><td colspan="2">target</td><td colspan="3">parameter</td></tr>
<tr><td rowspan="3">DESIGN-POINT</td><td colspan="2" rowspan="3">[inters]</td><td rowspan="3">sevent</td><td rowspan="3">variable</td><td>U-SPACE</td></tr>
<tr><td>V-SPACE</td></tr>
<tr><td>X-SPACE</td></tr>
</table>

PURPOSE:

Access specific values in the database and transmit them to the controlling process or write them to standard output if Proban runs by itself.

PARAMETERS:

U-SPACE	U-SPACE is used to calculate the u-space (standard normal space) values of all variables used in the definition of the specified event. Those values that are not specified in the command are set to their median value before calculation.
X-SPACE	X-SPACE is used to calculate the x-space (model space) values of all variables used in the definition of the specified event. Those values that are not specified in the command are set to their median value before calculation.
RESULT	Get a main result, a sensitivity factor or a design point value.
MAIN-RESULT	MAIN-RESULT is used to access any one main result. The result name <mresname> can be any of those allowed in the PRINT RESULT PARAMETER-STUDY command.
SENSITIVITY	SENSITIVITY is used to access any one sensitivity value (the derivative of target with respect to one parameter).
DESIGN-POINT	DESIGN-POINT is used to access the value of <variable> in the design point for single event <sevent>. The value is returned in either X-, V- or U-SPACE. The [inters] input is needed if a calculation of bounds was performed.
event	Name of event.
variable, xvalue	Matrix of variables and their corresponding x-space values.
variable, uvalue	Matrix of variables and their corresponding u-space values.

mresname	Main result name. Depends on the analysis type. For example Beta-FORM.
target	Depends on the analysis type. In a probability analysis the targets are 'Beta', 'Probability' and 'Log10-Prob' and in a distribution analysis the target names are 'Mean', 'Standard-Dev', 'Skewness' and 'Kurtosis'.
parameter	Name of parameter.
inters	Index of intersection if event is a union, for example 3.
variable	Name of variable for which the design point value is sought.
sevent	Name of single event.

NOTES:

Proban is prepared for use as a sub-process, controlled by another program which transmits commands to Proban. The GET command is used to access the calculated results. As an example one can invoke Proban from a code calibration program in order to calculate the reliability indexes required for the code calibration.

EXAMPLES:

```
GET U-SPACE Beam ( ONLY P1 30000 P2 30000 )
GET RESULT MAIN-RESULT Beta-FORM
GET RESULT SENSITIVITY Beta L1-Mean
GET RESULT DESIGN-POINT Beam L1 X-SPACE
```

HELP

<table>
<tr><td rowspan="12">HELP</td><td colspan="2">ABOUT-HELP</td></tr>
<tr><td colspan="2">COMMAND-INPUT-FILE</td></tr>
<tr><td rowspan="4">LINE-MODE</td><td>COMMANDS</td></tr>
<tr><td>DEFAULTS</td></tr>
<tr><td>SELECTING</td></tr>
<tr><td>OTHER-FACILITIES</td></tr>
<tr><td rowspan="4">PROGRAMMING-MODE</td><td>BUILT-IN</td></tr>
<tr><td>EXPRESSIONS</td></tr>
<tr><td>OVERVIEW</td></tr>
<tr><td>VARIABLES</td></tr>
<tr><td colspan="2">STATUS-LIST</td></tr>
<tr><td colspan="2">SUPPORT</td></tr>
</table>

PURPOSE:

Provide guidance to the user.

PARAMETERS:

ABOUT-HELP	Provide information about the HELP command.
COMMAND-INPUT-FILE	Provide information about command input files.
LINE-MODE	Provide information specific for usage in line mode.
COMMANDS	Provide information about specifying commands in line mode.
DEFAULTS	Provide information about usage of defaults in line mode.
SELECTING	Provide information about selection and abbreviation in line mode.
OTHER-FACILITIES	Provide information about special facilities in line mode.
PROGRAMMING-MODE	Provide information about the programming mode.
BUILT-IN	Provide information about built in functions, procedures and constants, accessible in programming mode.
EXPRESSIONS	Provide information about the use of calculation expressions in programming mode.

OVERVIEW	Provide an overview of the facilities available in programming mode.
VARIABLES	Provide information about the usage of variables in programming mode.
STATUS-LIST	Examine the status list for Proban.
SUPPORT	Provide information that is helpful at a support request. This include information about the versions of the program and linked-in libraries, and about the environment in which the program runs.

NOTES:

1 This command is not journalled.

2 There is no guarantee that this command will remain compatible over time.

3 All information, except the status list, is treated as a program message, i.e. it is written into the message window in graphics mode and echoed at the terminal in line mode. The status list is presented in the print window when running in graphics mode, and presented one full screen at a time when running in line mode.

4 See also the sections in Chapter 4 on getting help when running in line mode and in graphics mode.

PLOT

PLOT

PURPOSE:

Execute the last DISPLAY command and write the result to the currently selected plot file.

PARAMETERS:

None.

NOTES:

1 The plot file and format is specified by use of the SET PLOT command.

2 Note that the command does not actually write the display as seen on the screen to file - it re-executes the DISPLAY command, taking any changed settings into account.

3 This command is not available from the menu bar in graphics mode. Use FILE PLOT instead, or use the graphics pick mode.

See also:

- DISPLAY
- SET PLOT

EXAMPLES:

```
PLOT
```

PRINT

PRINT	ANALYSIS-SETTINGS	...
	CORRELATION	...
	DISTRIBUTION	...
	EVENT	...
	FUNCTION	...
	PARAMETER-STUDY	...
	RESULT	...
	STARTING-POINT	...
	VARIABLE	...

PURPOSE:

To present input data and results graphically.

PARAMETERS:

ANALYSIS-SETTINGS	Print all analysis settings related to probability and distribution analysis.
CORRELATION	Print all correlations assigned to a selection of variables.
DISTRIBUTION	Print the distribution and density functions of a variable.
EVENT	Print information about a selection of events.
FUNCTION	Print information about a function or a function value/derivative.
PARAMETER-STUDY	Print the assigned parameter study.
PLAN-INSPECTION	Print an inspection plan.
RESULT	Print an analysis result.
STARTING-POINT	Print the starting point assignment for a selection of events.
VARIABLE	Print information about a selection of variables.

NOTES:

The general print options can be manipulated through the SET PRINT command.

See also:

- SET PRINT

PRINT ANALYSIS-SETTINGS

...	ANALYSIS-SETTINGS

PURPOSE:

Print all analysis options.

PARAMETERS:

None.

NOTES:

All analysis options related to probability and distribution analysis are printed, including those for analysis methods that are currently not selected.

See also:

- DEFINE
- SELECT ANALYSIS-METHOD

EXAMPLES:

```
PRINT ANALYSIS-SETTINGS
```

PRINT CORRELATION

...	CORRELATION	univar+

PURPOSE:

Print assigned correlations.

PARAMETERS:

univar+ A selection of variables that are defined as one dimensional distributions with numerical or fixed parameters. All correlations assigned to pairs of these variables are printed. If only one variable is selected, all correlations assigned to this variable will be printed.

NOTES:

See also:

- ASSIGN CORRELATION
- SET TITLE

EXAMPLES:

```
PRINT CORRELATION *
```

Generates the following print:

```
                       +--------------------------------+
                       ! Correlations between variables !
                       +--------------------------------+

Variable 1   Variable 2   Input      Basic        Normalized
-----------------------------------------------------------------------------
J3220-lnA    J5120-lnA    Basic      CorrStress
```

PRINT DISTRIBUTION

...	DISTRIBUTION	univar+	LOW-RESOLUTION			
			HIGH-RESOLUTION	n		
			FRACTILE	{	probability	}*
			PROBABILITY	{	fractile	}*

PURPOSE:

Print distribution- and density functions and fractile values for the variables assigned distributions with fixed or numerical parameters.

PARAMETERS:

univar+	Selection of variables that are defined as one dimensional distributions with numerical or fixed parameters.
LOW-RESOLUTION	Print a table of the distribution, complementary distribution and density function values at 19 fixed probability values ranging from 0.001 to 0.999.
HIGH-RESOLUTION n	Print a table of the distribution, complementary distribution and density function values at n points ranging from median - 4 standard deviations to median + 4 standard deviations.
FRACTILE probability	Print fractile values at the specified probabilities. Also prints the complementary distribution and density function at the specified points.
PROBABILITY fractile	Print probabilities (distribution function values) at the specified fractiles. Also prints the complementary distribution and density function at the specified points.

NOTES:

If a LOOP is specified in line mode input after DISTRIBUTION, any specified fractiles or probabilities are kept as defaults. Otherwise, the default set of fractiles and probabilities is empty.

See also:

- DISPLAY DISTRIBUTION
- PRINT VARIABLE
- SET TITLE

EXAMPLES:

```
PRINT DISTRIBUTION StdNormal LOW-RESOLUTION
```

Generates the following print:

```
                         +---------------------------+
                         ! Distribution of StdNormal !
                         +---------------------------+

           Variable     Type         Parameter         Value
           -------------------------------------------------------
           StdNormal    Normal       Mean          0.000000000E+00
                                     Stand-Dev     1.000000000E+00

                 Fractile      Distr    Compl       Density
             ---------------------------------------------------
             -3.090232306E+00 0.001000 0.999000  3.367090077E-03
             -2.326347874E+00 0.010000 0.990000  2.665214220E-02

             -1.644853627E+00 0.050000 0.950000  1.031356404E-01
             -1.281551566E+00 0.100000 0.900000  1.754983319E-01
             -1.036433389E+00 0.150000 0.850000  2.331587753E-01
             -8.416212336E-01 0.200000 0.800000  2.799619204E-01
             -6.744897502E-01 0.250000 0.750000  3.177765727E-01
             -5.244005127E-01 0.300000 0.700000  3.476926142E-01
             -2.533471031E-01 0.400000 0.600000  3.863425335E-01
              5.293868432E-14 0.500000 0.500000  3.989422804E-01
              2.533471031E-01 0.600000 0.400000  3.863425335E-01
              5.244005127E-01 0.700000 0.300000  3.476926142E-01
              6.744897502E-01 0.750000 0.250000  3.177765727E-01
              8.416212336E-01 0.800000 0.200000  2.799619204E-01
              1.036433389E+00 0.850000 0.150000  2.331587753E-01
              1.281551566E+00 0.900000 0.100000  1.754983319E-01
              1.644853627E+00 0.950000 0.050000  1.031356404E-01
              2.326347874E+00 0.990000 0.010000  2.665214220E-02
```

PRINT EVENT

...	EVENT	name+

PURPOSE:

Print information about one or more events.

PARAMETERS:

name+	Name(s) of event(s) to be printed.

NOTES:

The printout contains information about the event data, including all assignments except starting point.

See also:

- CREATE EVENT
- CHANGE EVENT
- DISPLAY EVENT
- ASSIGN STARTING-POINT
- ASSIGN MEASURED-VALUE
- SET TITLE

EXAMPLES:

```
PRINT EVENT ( J3220-CGFail J5-I1-INAll )
```

Generates the following print:

```
               +-----------------------------------------------+
               !                     Event                     !
               +-----------------------------------------------+
               !                  J3220-CGFail                 !
               ! Crack growth failure for fatigue point J3220 !
               +-----------------------------------------------+

Event-type   Subevent     Subtype      Contents
-------------------------------------------------------------------------------
Single                                 J3220-CGFail < 0.0


                        +-----------------------------+
                        !            Event            !
                        +-----------------------------+
                        !         J5-I1-INAll         !
                        ! All inspections up to J5-I1 !
                        +-----------------------------+

Event-type   Subevent     Subtype      Contents
-------------------------------------------------------------------------------
Intersection J5-I1-INLen  Single       J5-I1-INLen = 0.0
```

PRINT FUNCTION

...	FUNCTION	DESCRIPTION	...
		FORMULA	...
		GRADIENT	...
		LIBRARY	...
		VALUE‾	...

PURPOSE:

Print information about the model functions that are available in the program.

PARAMETERS:

DESCRIPTION	Print a description of one or more functions.
FORMULA	Print a description and a calculation scheme for one of more function formulas.
GRADIENT	Calculate and print a gradient for a function.
LIBRARY	Print a description of a selection of function libraries.
VALUE	Calculate and print a gradient for a function.

NOTES:

None.

PRINT FUNCTION DESCRIPTION

...	DESCRIPTION	name+

PURPOSE:

Print a description of a selection of functions.

PARAMETERS:

<name>+ Name(s) of the function(s) to be printed.

NOTES:

The selection of functions presented is determined by the current selection of sub-libraries (see SELECT FUNCTION-LIBRARY). This is because some libraries may contain a large number of functions and/or not be relevant to the current problem.

See also:

- SELECT FUNCTION-LIBRARY
- PRINT FUNCTION-LIBRARY
- SET TITLE

EXAMPLES:

```
PRINT FUNCTION DESCRIPTION Difference
```

Generates the following print:

```
                            +---------------------+
                            !      Function       !
                            +---------------------+
                            !     Difference      !
                            ! Difference X1 - X2  !
                            +---------------------+

                   The function belongs to sublibrary: Misc

              First and second order derivatives are implemented


             Name         Description
----------------------------------------------------------------------------
Arguments:   Additive-Arg Additive argument
             Subtract-Arg Subtractive argument
```

PRINT FUNCTION FORMULA

...	FORMULA	name+

PURPOSE:

Print a description of a selection of function formulas.

PARAMETERS:

name+	Name(s) of the function formula(s) to be printed.

NOTES:

Prints the name, description, argument list, calculation scheme and definition of a function formula.

See also:

- CREATE FUNCTION FORMULA
- CHANGE FUNCTION FORMULA
- DELETE FUNCTION FORMULA
- RENAME FUNCTION FORMULA
- SET TITLE

EXAMPLES:

```
PRINT FUNCTION FORMULA SYMFUN
```

Generates the following print:

```
                            +---------------------+
                            !      Function       !
                            +---------------------+
                            !        SYMFUN       !
                            ! Symbolic Function   !
                            +---------------------+

                    Gradients must be calculated numerically

Name          DescriptionValue Index
----------------------------------------------------------------------------
A             Arg AV1
B             Arg BV2

Operator      OperandsResult
----------------------------------------------------------------------------
+             V1 V2V3

Formula: A+B
```

PRINT FUNCTION GRADIENT

<table><tr><td rowspan="3">...</td><td rowspan="3">GRADIENT</td><td rowspan="3">function</td><td rowspan="3">SINGLE-POINT</td><td>ANALYTICAL</td><td rowspan="3">[dim]</td><td rowspan="3">arguments</td></tr><tr><td>NUMERICAL</td></tr><tr><td>CHECK</td></tr></table>

PURPOSE:

Calculate and print the gradient of a function.

PARAMETERS:

function	Name of the function to be printed.
SINGLE-POINT	The gradient is to be calculated in a single point.
ANALYTICAL	Calculate only analytical gradients (i.e. those that are programmed into the function). This option is not available if the function cannot calculate gradients.
NUMERICAL	Calculate gradients by numerical differentiation only.
CHECK	Calculate both analytical and numerical gradients, and print both.
[dim]	The dimension of the value calculated by the function. Is not required as input if the dimension is fixed.
<arguments>	The arguments of the function.

NOTES:

1 The selection of functions presented is determined by the current selection of sub-libraries (see SELECT FUNCTION-LIBRARY). This is because some libraries may contain a large number of functions and/or not be relevant to the current problem.

2 If a LOOP is specified in line mode input after <function>, any specified argument values are kept as defaults. Otherwise, the default set of argument values is empty.

See also:

- SELECT FUNCTION-LIBRARY
- PRINT FUNCTION VALUE
- SET TITLE

EXAMPLES:

```
PRINT FUNCTION GRADIENT Power SINGLE-POINT CHECK 4 3
```

Generates the following print:

```
                          +-------------------------+
                          !        Function         !
                          +-------------------------+
                          !          Power          !
                          ! Power function: X1**X2  !
                          +-------------------------+

             Function     Argument          Value          Numerical
  ------------------------------------------------------------------------
  Argument:               Value         4.000000000E+00
                          Exponent      3.000000000E+00

  Function:  Power                      6.400000000E+01


  Gradient:  Power        Value         4.800000000E+01  4.804801600E+01
             Power        Exponent      8.872283911E+01  8.890758910E+01
```

PRINT FUNCTION LIBRARY

...	LIBRARY	name+

PURPOSE:

Print a description of a selection of function libraries.

PARAMETERS:

name+ Name(s) of the function libraries to be printed.

NOTES:

See also:

- SELECT FUNCTION-LIBRARY
- PRINT FUNCTION DESCRIPTION

EXAMPLES:

```
PRINT FUNCTION LIBRARY Misc
```

Generates the following print:

```
                                +---------------------------------+
                                !           Sublibrary            !
                                +---------------------------------+
                                !              Misc               !
                                ! Miscellaneous general functions !
                                +---------------------------------+

Function     Dimen  NArg NOp Description
---------------------------------------------------------------------------------
Difference       1     2   0 Difference X1 - X2
Division         1     2   0 Division X1 / X2
Identity         1     1   0 Identity: f(x) = x
Linear-Comb      1 Input   0 Linear combination: x1*x2 + x3*x4 + ...
Log-Diff         1     2   0 Difference: Log(X1) - Log(X2)
Maximum          1 Input   0 Maximum of any number of variables
Minimum          1 Input   0 Minimum of any number of variables
Polynom-1        1     4   0 Polynomium of degree 1
Polynom-2        1     5   0 Polynomium of degree 2
Polynom-3        1     6   0 Polynomium of degree 3
Polynom-4        1     7   0 Polynomium of degree 4
Polynom-N        1 Input   0 Polynomium(N,X,X0,C0,...): Sum of Ci*((X-X0)**i)
Power-Diff       1     3   0 Difference: X1**X3 - X2**X3
Product          1 Input   0 Product of any number of variables
SignPowDiff      1     3   0 Sign(X1)*(Abs(X1)**X3) - Sign(X2)*(Abs(X2)**X3)
Sum              1 Input   0 Sum of any number of variables
```

PRINT FUNCTION RESPONSESURFACE

...	RESPONSESURFACE	name+

PURPOSE:

Print description of a selection of response surface functions

PARAMETERS:

name+	Name(s) of the response surface function(s) to be printed.

NOTES:

Prints the name, description, argument list and definition of a response surface function.

See also:

- CREATE FUNCTION RESPONSESURFACE
- CHANGE FUNCTION RESPONSESURFACE
- DELETE FUNCTION
- RENAME FUNCTION
- SET TITLE

EXAMPLES:

```
PRINT FUNCTION FORMULA SYMFUN
```

Generates the following print:

```
                        +---------------------+
                        !       Function      !
                        +---------------------+
                        !        SYMFUN       !
                        ! Symbolic Function   !
                        +---------------------+

                Gradients must be calculated numerically

Name         DescriptionValue Index
-----------------------------------------------------------------------------
A            Arg AV1
B            Arg BV2

Operator     OperandsResult
-----------------------------------------------------------------------------
+            V1 V2V3

Formula: A+B
```

PRINT FUNCTION VALUE

...	VALUE	function	SINGLE-POINT	[dim]	argument*

PURPOSE:

Calculate and print the value of a function.

PARAMETERS:

function	Name of the function to be printed.
SINGLE-POINT	The value is to be calculated in a single point.
[dim]	The dimension of the value calculated by the function. Is not required as input if the dimension is fixed.
argument*	The arguments of the function.

NOTES:

1 The selection of functions presented is determined by the current selection of sub-libraries (see SELECT FUNCTION-LIBRARY). This is because some libraries may contain a large number of functions and/or not be relevant to the current problem.

2 If a LOOP is specified in line mode input after <function>, any specified argument values are kept as defaults. Otherwise, the default set of argument values is empty.

See also:

- SELECT FUNCTION-LIBRARY
- PRINT FUNCTION GRADIENT
- SET TITLE

EXAMPLES:

```
PRINT FUNCTION VALUE Polynomium-2 SINGLE-POINT 11 0 4 2 -6
```

Generates the following print:

```
                          +-------------------------+
                          !        Function         !
                          +-------------------------+
                          !       Polynom-2         !
                          ! Polynomium of degree 2  !
                          +-------------------------+

             Name         Value
-----------------------------------------------------------------------------
Arguments:   Argument      1.100000000E+01
             Shift         0.000000000E+00
             Coef-0        4.000000000E+00
             Coef-1        2.000000000E+00
             Coef-2       -6.000000000E+00

Function:    Polynom-2    -7.000000000E+02
```

PRINT PARAMETER-STUDY

...	PARAMETER-STUDY

PURPOSE:

Print the currently assigned parameter study.

PARAMETERS:

None.

NOTES:

See also:

- DEFINE PARAMETER-STUDY
- DEFINE ANALYSIS-OPTION PARAMETER-STUDY
- SET TITLE

EXAMPLES:

```
PRINT PARAMETER-STUDY
```

Generates the following print:

```
                         +--------------------------+
                         ! Assigned parameter study !
                         +--------------------------+


              Variable      Parameter     Number      Value
              -------------------------------------------------
              Nyears        Constant          10   2.000000000E+00
                                                   4.000000000E+00
                                                   6.000000000E+00
                                                   8.000000000E+00
                                                   1.000000000E+01
                                                   1.500000000E+01
                                                   2.000000000E+01
                                                   3.000000000E+01
                                                   5.000000000E+01
                                                   1.000000000E+02
```

PRINT RESULT

...	RESULT	ALL	...
		ANALYSIS-SETTINGS	...
		IMPORTANCE-FACTORS	...
		INTERMEDIATE-RESULTS	...
		PARAMETER-STUDY	...
		SAMPLE	...
		SENSITIVITY	...
		SUMMARY	...

PURPOSE:

Print the currently selected result in tabular form, to screen or to file.

PARAMETERS:

ALL	Print all the results from a probability or distribution analysis.
ANALYSIS-SETTINGS	Print analysis options applied to the result.
IMPORTANCE-FACTORS	Print the importance factors resulting from a probability analysis.
INTERMEDIATE-RESULTS	Print the intermediate (debug) results from a probability or distribution analysis.
PARAMETER-STUDY	Print results as a function of the parameter in a parameter study.
SAMPLE	Print the sample resulting from application of a simulation method.
SENSITIVITY	Print parametric sensitivity results.
SUMMARY	Print a summary of the results from a probability or distribution analysis.

NOTES:

None.

PRINT RESULT ALL

...	ALL	[value1+]	[value2+]

PURPOSE:

Print all information from the selected result.

PARAMETERS:

value1+	This input is only required if the selected result is a parameter study. value1 is then a selection of the parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.
value2+	This input is only required if the selected result is a two parameter study. value2 is then a selection of the parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.

NOTES:

The print does not contain the sample resulting from a simulation. This sample will often be very large, and it can be printed by use of PRINT RESULT SAMPLE.

See also:

- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT ALL          % no parameter study
DISPLAY RESULT ALL *      % all results from a study
```

PRINT RESULT ANALYSIS-SETTINGS

...	ANALYSIS-SETTINGS

PURPOSE:

Print analysis options applied to a probability, a crossing rate, a first passage probability or a distribution analysis.

PARAMETERS:

None.

NOTES:

This print contains the date/time and cpu time consumption for the analysis.

See also:

- DEFINE
- RUN CONTINUOUS-PROCESS-ANALYSIS
- RUN DETERMINISTIC-ANALYSIS
- RUN PROBABILITY-ANALYSIS
- RUN DISTRIBUTION-ANALYSIS
- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT ANALYSIS-SETTINGS
```

PRINT RESULT IMPORTANCE-FACTORS

...	IMPORTANCE-FACTORS	[value1+]	[value2+]

PURPOSE:

Print importance factors.

PARAMETERS:

value1+	This input is only required if the selected result is a parameter study. value1 is then a selection of the parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.
value2+	This input is only required if the selected result is a two parameter study. value2 is then a selection of the parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.

NOTES:

The smallest importance factor values may be removed from the print (see DEFINE PRESENTATION RESULT IMPORTANCE-CUTOFF).

See also:

- DEFINE PRESENTATION RESULT IMPORTANCE-CUTOFF
- PRINT RESULT PARAMETER-STUDY IMPORTANCE-FACTOR
- PRINT RESULT
- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT IMPORTANCE-FACTORS          % no parameter study
```

PRINT RESULT INTERMEDIATE-RESULTS

...	INTERMEDIATE-RESULTS

PURPOSE:

Print all intermediate results from the selected analysis result.

PARAMETERS:

None.

NOTES:

1 The intermediate results are generated during the analysis. The amount of intermediate results is controlled by use of the commands DEFINE ANALYSIS-OPTION INTERMEDIATE-RESULTS and DEFINE ANALYSIS-OPTIONS GENERATED-DISTRIBUTION INTERMEDIATE-RESULTS.

2 The print may be very long, depending on the amount of intermediate results requested.

3 The intermediate results form a parameter study cannot be selected separately. They will be printed in the order in which the parameter study was performed.

See also:

- DEFINE ANALYSIS-OPTION INTERMEDIATE-RESULTS
- DEFINE ANALYSIS-OPTION GENERATED-DISTRIBUTION INTERMEDIATE-RESULTS
- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT INTERMEDIATE-RESULTS
```

PRINT RESULT PARAMETER-STUDY

...	PARAMETER-STUDY	IMPORTANCE-FACTOR	...
		MAIN-RESULT	...

PURPOSE:

Print results as a function of the parameter in a parameter study.

PARAMETERS:

IMPORTANCE-FACTOR	Print importance factors as a function of the parameter.
MAIN-RESULT	Print one or more main results as a function of the parameter.

NOTES:

None.

PRINT RESULT PARAMETER-STUDY IMPORTANCE-FACTOR

...	IMPORTANCE-FACTOR	impname+

PURPOSE:

Print importance factors as a function of the parameter in a parameter study.

PARAMETERS:

impname+	A selection of importance factor names. The segment named "Other" in the pie chart representation is not used here. All available importance factor names can be selected.

NOTES:

See also:

- DISPLAY RESULT PARAMETER-STUDY IMPORTANCE-FACTOR
- PRINT RESULT IMPORTANCE-FACTORS
- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT PARAMETER-STUDY IMPORTANCE-FACTOR ( ONLY Depth ImpGroup-1 )
PRINT RESULT PARAMETER-STUDY IMPORTANCE-FACTOR ONLY T*
```

PRINT RESULT PARAMETER-STUDY MAIN-RESULT

...	MAIN-RESULT	mainres+

PURPOSE:

Print main results as a function of the parameter in a parameter study.

PARAMETERS:

mainres+	A selection of main results. The list of available results depend on the analysis performed. All possible main results are presented in the list, even though they may not all be calculated for all the individual analyses in the parameter study. For deterministic analysis of a variable there will be one result for each coordinate, and for an event there will be one result. These results will be named after the variable or event analysed.
coordinate+	A coordinate of a vector if a vector variable with more than one coordinate is sampled.

NOTES:

None.

See also:

- DISPLAY RESULT PARAMETER-STUDY MAIN-RESULT
- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT PARAMETER-STUDY MAIN-RESULT ( ONLY Prob* Conf* )
PRINT RESULT PARAMETER-STUDY MAIN-RESULT ONLY *Mean*
```

PRINT RESULT SAMPLE

<table>
<tr><td rowspan="5">...</td><td rowspan="5">SAMPLE</td><td rowspan="5">[value1+]</td><td rowspan="5">[value2+]</td><td colspan="4">LOW-RESOLUTION</td></tr>
<tr><td colspan="2">HIGH-RESOLUTION</td><td colspan="2">n</td></tr>
<tr><td colspan="4">ALL-SIMULATIONS</td></tr>
<tr><td>FRACTILE</td><td>{</td><td>probability</td><td>}*</td></tr>
<tr><td>PROBABILITY</td><td>{</td><td>fractile</td><td>}*</td></tr>
</table>

PURPOSE:

Print distribution- and density functions and fractile values for the variables assigned distributions with fixed or numerical parameters.

PARAMETERS:

value1+	This input is only required if the selected result is a parameter study. Value1 is then a selection of the first-parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.
value2+	This input is only required if the selected result is a two-parameter study. Value2 is then a selection of the second-parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.
LOW-RESOLUTION	Print a table of the distribution, complementary distribution and density function values at 19 fixed probability values ranging from 0.001 to 0.999.
HIGH-RESOLUTION n	Print a table of the distribution, complementary distribution and density function values at n points ranging from median - 4 standard deviations to median + 4 standard deviations.
ALL-SIMULATIONS	The sampled values are printed in sorted order (increasing probability.)
FRACTILE probability	Print fractile values at the specified probabilities. Also prints the complementary probabilities.
PROBABILITY fractile	Print probabilities (distribution function values) at the specified fractiles. Also prints the complementary probabilities at the specified points.

NOTES:

If a LOOP is specified in line mode input after SAMPLE, any specified fractiles or probabilities are kept as defaults. Otherwise, the default set of fractiles and probabilities is empty.

See also:

- PRINT DISTRIBUTION
- SET TITLE

EXAMPLES:

```
PRINT RESULT SAMPLE FRACTILE ( ONLY 0.5 0.6 )
```

Generates the following print:

```
        +----------------------------------------------------------------+
        !                   Network Planning Example                     !
        +----------------------------------------------------------------+
        ! Distribution of  : Network-Longest                             !
        !                    Duration of all paths through the network !
        ! Analysis method  : Latin Hypercube simulation                  !
        +----------------------------------------------------------------+

                 SAMPLED DISTRIBUTION
                 CALCULATED FRACTILES

          Fractile           Distr            Compl
      --------------------------------------------------
       6.791248101E+01  5.000000000E-01  5.000000000E-01
       6.899417939E+01  6.000000000E-01  4.000000000E-01
```

PRINT RESULT SENSITIVITY

...	SENSITIVITY	[value1+]	[value2+]	coordinate

PURPOSE:

Print the parametric sensitivity values for the selected result.

PARAMETERS:

value1+	This input is only required if the selected result is a parameter study. Value1 is then a selection of the first-parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.
value2+	This input is only required if the selected result is a two-parameter study. Value2 is then a selection of the second-parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.
coordinate+	A coordinate of a vector if a vector vareiable with more than one coordinate is sampled.

NOTES:

1 The sensitivity values are printed for the probability itself, the logarithm of the probability and for the reliability index.

2 The sensitivity measure is calculated as the change in the target value resulting from a fixed percentage increase in the parameter. This value provides a dimensionless sensitivity measure. The definition of the sensitivity measure can be changed using the command:

See also:

- DEFINE PRESENTATION RESULT SENSITIVITY-MEASURE.
- ASSIGN SENSITIVITY
- DEFINE ANALYSIS-OPTION SENSITIVITY
- SELECT RESULT
- SET TITLE

EXAMPLES:

```
PRINT RESULT SENSITIVITY
```

may generate the following print:

```
                   +---------------------------------------+
                   ! Probability of : Fatigue < 0.0        !
                   !                  Fatigue Life, SN II  !
                   ! Analysis method: SORM                 !
                   +---------------------------------------+


       Parametric sensitivity result for Probability = 1.67162275386E-08

   Variable     Type         Parameter      Value    dProb/dPar  Measure
   ----------------------------------------------------------------------
   Scale        Normal       Mean          5.048E+00  7.229E-08  3.65E-08
                             Stand-Dev     6.000E-01  1.815E-07  1.09E-08

             Parametric sensitivity result for Beta = 5.5224397018

   Variable     Type         Parameter      Value    dBeta/dPar  Measure
   ----------------------------------------------------------------------
   Scale        Normal       Mean          5.048E+00 -7.596E-01  -0.38347
                             Stand-Dev     6.000E-01 -1.907E+00  -0.11442

         Parametric sensitivity result for Log10(Prob) = -7.7768617259

   Variable     Type         Parameter      Value    dLg10/dPar  Measure
   ----------------------------------------------------------------------

   Scale        Normal       Mean          5.048E+00  1.878E+00   0.94814
                             Stand-Dev     6.000E-01  4.715E+00   0.28290
```

PRINT RESULT SUMMARY

...	SUMMARY	[value+]

PURPOSE:

Print a short summary for the selected result.

PARAMETERS:

value+	This input is only required if the selected result is a parameter study. <value> is then a selection of the parameter values for which the study was run. The particular results from the analysis using the selected value(s) will be printed.

NOTES:

See also:

- SELECT RESULT
- SET PRINT

EXAMPLES:

```
PRINT RESULT SUMMARY
```

may generate the following print:

```
  +---------------------------------------+
  !        Fatigue by Wave Loading        !
  +---------------------------------------+
  ! Probability of : Fatigue < 0.0        !
  !                  Fatigue Life, SN II  !
  ! Analysis method: SORM                 !
  +---------------------------------------+

FORM Probability:                  1.72486E-08
SORM Probability:                  1.67162E-08

FORM Reliability index:                 5.5169
SORM Reliability index:                 5.5224
```

PRINT VARIABLEPRINT VARIABLE

...	VARIABLE	name+

PURPOSE:

Print information about one or more variables.

PARAMETERS:

name+	Name(s) of variable(s) to be printed.

NOTES:

The printout contains information about the variable data, including all assignments except starting point and correlation.

See also:

- CREATE VARIABLE
- CHANGE VARIABLE
- DISPLAY VARIABLE
- ASSIGN CONDITIONING
- ASSIGN EXTREME-VALUE
- ASSIGN FUNCTION-OPTION
- ASSIGN OPTIMISATION-BOUNDS
- ASSIGN SENSITIVITY-CALCULATION
- SET TITLE

EXAMPLES:

```
PRINT VARIABLE ( J3220-lnC J3220-m )
```

May generate the following print:

```
                         +----------------------------+
                         !          Variable          !
                         +----------------------------+
                         !          J3220-lnC         !
                         ! ln(C), material parameter !
                         +----------------------------+

Type         Name             Dim Parameter    Value                    Sens
-----------------------------------------------------------------------------
Distribution Normal             1 Mean         -31.0                    Off
                                  Stand-Dev    0.77                     Off

Calculated parameters:            Skewness     0.0
                                  Kurtosis     3.0
                                  Median       -31.0


                           +------------------------+
                           !        Variable        !
                           +------------------------+
                           !         J3220-m        !
                           ! m, material parameter !
                           +------------------------+

Type         Name             Dim Parameter    Value                    Sens
-----------------------------------------------------------------------------
Fixed                                          3.5
```

RENAME

RENAME	EVENT	...
	FUNCTION	...
	RESULT	...
	VARIABLE	...

PURPOSE:

Rename a named object.

PARAMETERS:

EVENT	Rename an event.
FUNCTION	Rename a function formula.
RESULT	Rename an analysis result.
VARIABLE	Rename a random variable.

NOTES:

None.

RENAME EVENT

...	EVENT	from	to

PURPOSE:

To change the name of an event.

PARAMETERS:

from	The original name of the event.
to	The new name of the event. This cannot be the name of an existing event.

NOTES:

Renaming of an event does not affect the usage of the event in other events.

See also:

- CHANGE EVENT
- CREATE EVENT
- DELETE EVENT
- COPY EVENT
- PRINT EVENT
- DISPLAY EVENT

EXAMPLES:

```
RENAME EVENT Moment-1 Moment-2
```

RENAME FUNCTION

...	FUNCTION	from	to

PURPOSE:

To change the name of a function formula or function integral.

PARAMETERS:

from	The original name of the function.
to	The new name of the function. This cannot be the name of an existing function.

NOTES:

If the renamed function is referenced in other function formulas or function integrals, then the name must be changed in these functions too.

See also:

- CHANGE FUNCTION
- CREATE FUNCTION
- DELETE FUNCTION
- PRINT FUNCTION
- DISPLAY FUNCTION

EXAMPLES:

```
RENAME FUNCTION SYMFUN SYMFOR
```

RENAME RESULT

...	RESULT	from	to

PURPOSE:

To change the name of a result.

PARAMETERS:

from	The original name of the result.
to	The new name of the result. This cannot be the name of an existing result.

NOTES:

See also:

- SAVE RESULT
- DELETE RESULT
- RUN
- PRINT RESULT
- DISPLAY RESULT

EXAMPLES:

```
RENAME RESULT SORM-Result Global-Fail
```

RENAME VARIABLE

...	VARIABLE	from	to

PURPOSE:

To change the name of a variable.

PARAMETERS:

from — The original name of the variable.

to — The new name of the variable. This cannot be the name of an existing variable.

NOTES:

Renaming a variable does not affect the usage of the variable in other variables or in single events, nor does it affect any correlation assignments.

See also:

- CHANGE VARIABLE
- CREATE VARIABLE
- DELETE VARIABLE
- COPY VARIABLE
- PRINT VARIABLE
- DISPLAY VARIABLE

EXAMPLES:

```
RENAME VARIABLE Width1 Width2
```

RUN

RUN	CONTINUOUS-PROCESS-ANALYSIS	...
	DETERMINISTIC-ANALYSIS	...
	DISTRIBUTION-ANALYSIS	...
	INPUT-CHECK	...
	PROBABILITY-ANALYSIS	...
	RESTART	...

PURPOSE:

Run an analysis.

PARAMETERS:

CONTINUOUS-PROCESS-ANALYSIS	Run a first passage probability analysis or a crossing rate analysis.
DETERMINISTIC-ANALYSIS	Run a deterministic analysis.
DISTRIBUTION-ANALYSIS	Run an analysis of the distribution of a variable.
INPUT-CHECK	Check the input for a probability analysis or distribution analysis.
INSPECTION-ANALYSIS	Run an analysis of the probability of failure for a fatigue point throughout the service life, taking all inspections into account.
PROBABILITY-ANALYSIS	Run an analysis of the probability of an event, possibly conditioned on another event, or of the probability of failure for a fatigue point throughout the service life.
RESTART	Restart a probability or distribution simulation from the results obtained.

NOTES:

None.

RUN CONTINUOUS-PROCESS-ANALYSIS

...	CONTINUOUS-PROCESS-ANALYSIS	CROSSING-RATE	...
		FIRST-PASSAGE-PROBABILITY	...

PURPOSE:

Run an analysis.

PARAMETERS:

CROSSING-RATE	Run a crossing rate analysis.
FIRST-PASSAGE-PROBABILITY	Run a first passage probability analysis.

NOTES:

None.

RUN CONTINUOUS-PROCESS-ANALYSIS CROSSING-RATE

...	CROSSING-RATE	event			
		SINGLE-EVENT	1d-variable	< , >	threshold

PURPOSE:

Run a crossing-rate analysis.

PARAMETERS:

event	Name of event to be analysed. The event cannot be a conditional event or contain equality events.
SINGLE-EVENT	Event is specified directly as a simple inequality.
1d-variable	Name of a one-dimensional variable (can be a coordinate of a multidimensional variable).
< , >	One of: < less than, > greater than
threshold	Numerical right hand side of the single event.

NOTES:

1 The type of analysis being run is selected by use of the SELECT ANALYSIS-METHOD CROSSING-RATE-ANALYSIS command. The options to be used for the analysis are set by use of the DEFINE command.

2 The result is stored under the name “LastAnalysis” and is overwritten the next time an analysis is performed unless saved under another name using the SAVE RESULT command.

3 The results are examined by use of the commands PRINT RESULT or DISPLAY RESULT.

4 Variables with type attribute PROBABILITY cannot be used in a crossing rate analysis.

See also:

- DEFINE ANALYSIS-OPTION
- DEFINE FORM-SORM
- DEFINE PARAMETER-STUDY
- SELECT ANALYSIS-METHOD CROSSING-RATE-ANALYSIS
- SAVE RESULT
- PRINT RESULT

- DISPLAY RESULT

EXAMPLES:

```
RUN CONTINUOUS-PROCESS-ANALYSIS CROSSING-RATE Cross_Ev
RUN CONTINUOUS-PROCESS-ANALYSIS CROSSING-RATE SINGLE-EVENT Cross_Var > 50
```

RUN CONTINUOUS-PROCESS-ANALYSIS FIRST-PASSAGE-PROBABILITY

...	FIRST-PASSAGE-PROBABILITY	event			
		SINGLE-EVENT	1d-variable	< , >	threshold

PURPOSE:

Run a first-passage probability analysis.

PARAMETERS:

event	Name of the event to be analysed. The event cannot be a conditional event or contain equality events.
SINGLE-EVENT	Event is specified directly as a simple inequality.
1d-variable	Name of a one-dimensional variable (can be a coordinate of a multidimensional variable).
< , >	One of: < less than, > greater than
threshold	Numerical right hand side of the single event.

NOTES:

1 The type of analysis being run is selected by use of the SELECT ANALYSIS-METHOD FIRST-PASSAGE-PROBABILITY-ANALYSIS command. The options to be used for the analysis are set by use of the DEFINE command.

2 The result is stored under the name “LastAnalysis” and is overwritten the next time an analysis is performed unless saved under another name using the SAVE RESULT command.

3 The results are examined by use of the commands PRINT RESULT or DISPLAY RESULT.

4 Variables with type attribute PROBABILITY cannot be used in a crossing rate analysis.

See also:

- DEFINE ANALYSIS-OPTION
- DEFINE FORM-SORM
- DEFINE PARAMETER-STUDY
- SELECT ANALYSIS-METHOD FIRST-PASSAGE-PROBABILITY-ANALYSIS
- SAVE RESULT

- PRINT RESULT
- DISPLAY RESULT

EXAMPLES:

```
RUN CONTINUOUS-PROCESS-ANALYSIS FIRST-PASSAGE-PROBABILITY FP_Ev
RUN CONTINUOUS-PROCESS-ANALYSIS FIRST-PASSAGE-PROBABILITY SINGLE-EVENT FP_Var > 50
```

RUN DETERMINISTIC-ANALYSIS

...	DETERMINISTIC-ANALYSIS	...

<table>
<tr><td rowspan="6">...</td><td rowspan="4">VARIABLE</td><td rowspan="4">variable</td><td colspan="6">MEAN-VALUE</td></tr>
<tr><td colspan="6">MEDIAN-VALUE</td></tr>
<tr><td rowspan="2">MODIFIED</td><td>MEAN-BASED</td><td rowspan="2">{</td><td rowspan="2">name</td><td rowspan="2">value</td><td rowspan="2">}*</td></tr>
<tr><td>MEDIAN-BASED</td></tr>
<tr><td rowspan="2">EVENT</td><td rowspan="2">event</td><td colspan="6">STARTING-POINT</td></tr>
<tr><td colspan="6">USPACE-ORIGIN</td></tr>
</table>

PURPOSE:

Run a deterministic analysis.

PARAMETERS:

VARIABLE	Calculate the value of a variable.
variable	The name of the variable for which the analysis is made.
MEAN-VALUE	Use the mean value of all distribution variables (the median will be used if the mean cannot be calculated).
MEDIAN-VALUE	Use the median value (50% fractile) of all distributions
MODIFIED	Use the mean or median as basis
MEAN-BASED	Use the mean value of all distribution variables as basis, excepting the specified modifications (the median will be used if the mean cannot be calculated).
MEDIAN-BASED	Use the median value (50% fractile) of all distributions as basis, excepting the specified modifications.
{name, value}*	Input of values that are to overwrite values specified elsewhere. Name is a one-dimensional variable of distribution type and value is either a single numerical value or -fracxx, the fractile at xx% probability level, followed by a numerical value. The -frac is case insensitive. Notice the preceding hyphen.
EVENT	Calculate the limit state value of an event. The limit state value is: left hand side - right hand side for a single event, minimum of all subevent values for an intersection, maximum of all subevent values for a union.Conditional events cannot be used here.
event	The name of the event for which an analysis is made.

STARTING-POINT	Use the starting point for the event, if assigned. If no starting point is assigned, the default starting point is used.
USPACE-ORIGIN	Calculate the value at the U-space origin (identical to the median values).

NOTES:

1 The result is stored under the name “LastAnalysis” and is overwritten the next time an analysis is performed unless saved under another name using the SAVE RESULT command.

2 The results are examined by use of the commands PRINT RESULT or DISPLAY RESULT.

See also:

- ASSIGN STARTING-POINT
- SAVE RESULT
- PRINT RESULT
- DISPLAY RESULT

EXAMPLES:

```
RUN DETERMINISTIC-ANALYSIS VARIABLE P-SNTime MEAN-VALUE
```

RUN DISTRIBUTION-ANALYSIS

...	DISTRIBUTION-ANALYSIS	...

<table>
<tr><td rowspan="3">...</td><td colspan="6">1d-variable</td></tr>
<tr><td rowspan="2">CONDITIONED</td><td rowspan="2">1d-variable</td><td colspan="4">event</td></tr>
<tr><td>SINGLE-EVENT</td><td>1d-condvar</td><td>< , ></td><td>threshold</td></tr>
</table>

PURPOSE:

Run a distribution analysis.

PARAMETERS:

1d-variable	The name of a one-dimensional variable (can be a coordinate of a multidimensional variable).
CONDITIONED	Analyse the conditioned distribution of <1d-variable> given an event.
event	The name of the conditioning event. This event cannot be of the conditioned type.
SINGLE-EVENT	The conditioning event is specified directly as a simple (in)equality.
1d-condvar	The name of the one dimensional variable that is forming the left hand side if the (in)equality.
< , >	One of: < less than, > greater than.
threshold	The numerical right hand side of the conditioning single event.

NOTES:

1 The type of analysis being run is selected by use of the SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS command. The options to be used for the analysis are set by use of the DEFINE command.

2 The result is stored under the name “LastAnalysis” and is overwritten the next time an analysis is performed unless saved under another name using the SAVE RESULT command.

3 The results are examined by use of the commands PRINT RESULT or DISPLAY RESULT.

See also:

- DEFINE DISTRIBUTION-SIMULATION

- DEFINE MEAN-VALUE-FORM
- DEFINE PARAMETER-STUDY
- SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS
- SAVE RESULT

EXAMPLES:

```
RUN DISTRIBUTION-ANALYSIS NPV
RUN DISTRIBUTION-ANALYSIS CONDITIONED NPV SINGLE-EVENT EXPENSE > 100000
```

RUN INPUT-CHECK

...	INPUT-CHECK	DISTRIBUTION-ANALYSIS
		CONTINUOUS-PROCESS-ANALYSIS
		PROBABILITY-ANALYSIS

PURPOSE:

Run a check of the input to an analysis.

PARAMETERS:

CONTINUOUS-PROCESS-ANALYSIS	Run a check of an analysis of the first passage probability or crossing rate of a variable.
DISTRIBUTION-ANALYSIS	Run a check of an analysis of the distribution of a variable.
PROBABILITY-ANALYSIS	Run a check of an analysis of the probability of an event, possibly conditioned on another event.

NOTES:

The sub-commands are identical in syntax to RUN CONTINUOUS-PROCESS-ANALYSIS, RUN DISTRIBUTION-ANALYSIS and RUN PROBABILITY-ANALYSIS. The only difference is that they only check the input to the analysis, they do not run the analysis.

RUN PROBABILITY-ANALYSIS

...	PROBABILITY-ANALYSIS	event			
		SINGLE-EVENT	1d-variable	< , = , >	threshold
		CONDITIONED	...		

...	event	...		
	SINGLE-EVENT	1d-variable	< , = , >	threshold

...	conditioning event			
	SINGLE-EVENT	1d-variable	< , = , >	threshold

PURPOSE:

Run a probability analysis.

PARAMETERS:

event	The name of the event to be analysed.
SINGLE-EVENT	The event is specified directly as a simple (in)equality.
1d-variable	The name of a one-dimensional variable (can be a coordinate of a multidimensional variable).
< , = , >	One of: < less than, = equal, > greater than.
threshold	The numerical right hand side of the single event.
CONDITIONED	Analyse the conditioned probability of one event given another.
conditioning event	The name of the conditioning event. This event cannot be of the conditioned type.

NOTES:

1 The type of analysis being run is selected by use of the SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS command. The options to be used for the analysis are set by use of the DEFINE command.

2 The result is stored under the name “LastAnalysis” and is overwritten the next time an analysis is performed unless saved under another name using the SAVE RESULT command.

3 The results are examined by use of the commands PRINT RESULT or DISPLAY RESULT.

See also:

- DEFINE ANALYSIS-OPTION
- DEFINE FORM-SORM
- DEFINE PROBABILITY-SIMULATION
- DEFINE PARAMETER-STUDY
- SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS
- SAVE RESULT
- PRINT RESULT
- DISPLAY RESULT

EXAMPLES:

```
RUN PROBABILITY-ANALYSIS Beam-Fail
RUN PROBABILITY-ANALYSIS SINGLE-EVENT NPV > 100000
RUN PROBABILITY-ANALYSIS CONDITIONED Failure NoFind
RUN PROBABILITY-ANALYSIS CONDITIONED Loss SINGLE-EVENT Expense > 100000
```

RUN RESTART

...	RESTART

PURPOSE:

Continue a simulation.

PARAMETERS:

None.

NOTES:

1 The selected result defines the analysis to be restarted.

2 Only simulations resulting from RUN PROBABILITY-ANALYSIS or RUN DISTRIBUTION-ANALYSIS can be restarted. The simulations will add to the previously established sample. The stop criteria for the simulation can be modified before the analysis is restarted.

3 The new result will be stored under the default name “LastAnalysis”. The previous result is deleted if it was also stored under this name.

See also:

- RUN DISTRIBUTION-ANALYSIS
- RUN PROBABILITY-ANALYSIS
- DEFINE DISTRIBUTION-SIMULATION
- DEFINE PROBABILITY-SIMULATION
- SAVE RESULT
- PRINT RESULT
- DISPLAY RESULT

EXAMPLES:

```
RUN RESTART
```

SAVE

SAVERESULT	...

PURPOSE:

Save an analysis result under a name.

PARAMETERS:

RESULT	Save an analysis result.

NOTES:

None.

SAVE RESULT

...	RESULT	name	desc

PURPOSE:

Save a result under a name.

PARAMETERS:

name	Name of the result. This cannot be the name of an existing result. Result names are matched case insensitive and can not be longer than 12 characters.
desc	Descriptive text for the result. It can be up to 50 characters long.

NOTES:

1 Only results from RUN DETERMINISTIC-ANALYSIS, RUN PROBABILITY-ANALYSIS and RUN DISTRIBUTION-ANALYSIS can be saved using this command. These results are by default stored under the name “LastAnalysis” and will be overwritten by the next analysis if they are not saved.

2 The results created by this program should not be modified by the user.

See also:

- RUN DISTRIBUTION-ANALYSIS
- RUN PROBABILITY-ANALYSIS
- DELETE RESULT
- RENAME RESULT
- DISPLAY RESULT
- PRINT RESULT

EXAMPLES:

```
SAVE RESULT Fail-444S 'SORM: Failure of joint 444'
```

SELECT

SELECT	ANALYSIS-METHOD	...
	FUNCTION-LIBRARY	...
	RESULT	...

PURPOSE:

Select objects or methods for use in other commands.

PARAMETERS:

ANALYSIS-METHOD	Select a method for use in probability and distribution analysis.
FUNCTION-LIBRARY	Select the function libraries to be available in other commands.
RESULT	Select the result to be used for presentation (PRINT/DISPLAY).

NOTES:

None.

SELECT ANALYSIS-METHOD

<table>
<tr><td>...</td><td>ANALYSIS-METHOD</td><td>...</td></tr>
</table>

<table>
<tr><td rowspan="16">...</td><td>CROSSING-RATE-ANALYSIS</td><td colspan="2">FORM</td></tr>
<tr><td rowspan="3">DISTRIBUTION-ANALYSIS</td><td colspan="2">MONTE-CARLO-SIMULATION</td></tr>
<tr><td colspan="2">LATIN-HYPERCUBE-SIMULATION</td></tr>
<tr><td colspan="2">MEAN-VALUE-FORM</td></tr>
<tr><td colspan="2">FIRST-PASSAGE-PROBABILITY-ANALYSIS</td><td>FORM</td></tr>
<tr><td rowspan="11">PROBABILITY-ANALYSIS</td><td colspan="2">FORM</td></tr>
<tr><td rowspan="4">SORM</td><td>PARABOLIC</td></tr>
<tr><td>DIAGONAL</td></tr>
<tr><td>FULL-EXPANSION</td></tr>
<tr><td>ASYMPTOTIC</td></tr>
<tr><td colspan="2">AXIS-ORTHOGONAL-SIMULATION</td></tr>
<tr><td rowspan="2">DESIGN-POINT-SIMULATION</td><td>DEFAULT</td></tr>
<tr><td>ADJUSTED</td></tr>
<tr><td colspan="2">DIRECTIONAL-SIMULATION</td></tr>
<tr><td rowspan="2">MONTE-CARLO-SIMULATION</td><td>CENTRAL-NORMAL</td></tr>
<tr><td>ADJUSTED</td></tr>
</table>

PURPOSE:

Select analysis method for probability and distribution analyses.

PARAMETERS:

CROSSING-RATE-ANALYSIS	Select the method used for crossing-rate analysis.
FIRST-PASSAGE-PROBABILITY-ANALYSIS	Select the method used for first-passage probability analysis.
DISTRIBUTION-ANALYSIS	Select the method used for distribution analysis.
MONTE-CARLO-SIMULATION	The simplest simulation method where points are picked randomly and sample values are kept (distribution analysis) or the frequency of occurrences counted (probability analysis).

LATIN-HYPERCUBE-SIMULATION	A stratified simulation technique where the sampling points are spread systematically over the sample space.
MEAN-VALUE-FORM	A simple FORM estimation of a distribution. Quick, but not generally reliable.
PROBABILITY-ANALYSIS	Select the method used for probability analysis.
FORM	First Order Reliability method.
SORM	Second Order Reliability Method.
PARABOLIC	Uses a parabolic approximation to the failure surface. If the U-space dimension is n, this method requires (n-1)2 second order derivations.
DIAGONAL	Uses an approximation to the failure surface based on the diagonal of the second order differential matrix. If the U-space dimension is n, this method requires n second order derivations.
FULL-EXPANSION	Uses a full second order approximation to the failure surface. If the U-space dimension is n, this method requires n2 second order derivations. Note that this method is not invariant art. different formulations of the problem that give the same failure surface.
ASYMPTOTIC	Asymptotic second order approximation. Not necessarily accurate, but fast.
AXIS-ORTHOGONAL-SIMULATION	A simulation method based on a FORM result. It simulates the difference between the correct probability and the FORM approximation.
DESIGN-POINT-SIMULATION	Design point simulation of probability. Monte Carlo sampling of points around the design point.
DIRECTIONAL-SIMULATION	Directional simulation of probability. Samples directions in U-space instead of points.
MONTE-CARLO-SIMULATION	Monte Carlo simulation of probability.
CENTRAL-NORMAL	The simulation density is entered at the u-space origin.
ADJUSTED	The simulation density incorporates variables assigned as adjusted simulation density in a sampling of probability. The sampling adjustment is for the standard normal u-space variables and is restricted to normal random variables.

NOTES:

1 The current analysis selection may be printed by use of the PRINT ANALYSIS-SETTINGS command.

2 Both a probability and a distribution analysis method is selected at the same time.

See also:

- PRINT ANALYSIS-SETTINGS
- DEFINE ANALYSIS-OPTIONS
- DEFINE CONTINUOUS-PROCESS
- DEFINE DISTRIBUTION-SIMULATION
- DEFINE MEAN-VALUE-FORM
- DEFINE DISTRIBUTION-SIMULATION
- DEFINE PROBABILITY-SIMULATION
- RUN PROBABILITY-ANALYSIS
- RUN CONTINUOUS-PROCESS ANALYSIS
- RUN DISTRIBUTION-ANALYSIS
- ASSIGN SIMULATION-DENSITY

EXAMPLES:

The following values are default when the program starts up with a new database:

```
SELECT ANALYSIS-METHOD PROBABILITY-ANALYSIS FORM
SELECT ANALYSIS-METHOD DISTRIBUTION-ANALYSIS MONTE-CARLO-SIMULATION
```

SELECT FUNCTION-LIBRARY

...	FUNCTION-LIBRARY	name+

PURPOSE:

Select one or more function libraries in order to limit the selection of functions presented in other commands.

PARAMETERS:

name+	A selection of function library names.

NOTES:

1 This command serves to mask off some function libraries temporarily. This can be useful as some function libraries may have a large number of functions and/or be irrelevant for the current modelling.

2 The program starts on a new database with two libraries masked off: “Distribution” and “Verification”.

See also:

- PRINT FUNCTION
- CREATE VARIABLE ... FUNCTION
- CHANGE VARIABLE ... FUNCTION

EXAMPLES:

```
SELECT FUNCTION-LIBRARY *
```

SELECT RESULT

...	RESULT	name

PURPOSE:

Select a result from probability, crossing rate, first passage probability, distribution analysis or deterministic analysis for presentation.

PARAMETERS:

name	The name of a result.

NOTES:

Only one analysis result can be presented at one time. Other types of result presentations are not affected by this command.

See also:

- PRINT RESULT
- DISPLAY RESULT
- DELETE RESULT
- RUN CONTINUOUS-PROCESS-ANALYSIS
- RUN DETERMINISTIC-ANALYSIS
- RUN DISTRIBUTION-ANALYSIS
- RUN PROBABILITY-ANALYSIS
- SAVE RESULT

EXAMPLES:

```
SELECT RESULT Fail-444S
```

SET

SET	COMPANY-NAME	...
	DISPLAY	...
	DRAWING	...
	GRAPH	...
	PLOT	...
	PRINT	...

PURPOSE:

Set or re-set global file/device environment characteristics.

PARAMETERS:

COMPANY-NAME	Set company name on display and plot.
DISPLAY	Set display characteristics.
DRAWING	Set drawing characteristics.
GRAPH	Set graph characteristics.
PLOT	Set plot file characteristics.
PRINT	Set print characteristics.

NOTES:

All sub-commands and data are fully explained subsequently as each command is described in detail.

SET COMPANY-NAME

...	COMPANY-NAME	text

PURPOSE:

To set the company name for use with result presentation.

PARAMETERS:

text The name of the company.

NOTES:

The text is used at the top of a display/plot. It is not used with printed results.

See also:

- DISPLAY
- PLOT

EXAMPLES:

```
SET COMPANY-NAME 'Det Norske Veritas'
```

SET DISPLAY

<table>
<tr><td rowspan="7">...</td><td rowspan="7">DISPLAY</td><td rowspan="2">COLOUR</td><td colspan="4">ON</td></tr>
<tr><td colspan="4">OFF</td></tr>
<tr><td rowspan="2">DESTINATION</td><td colspan="4">FILE</td></tr>
<tr><td colspan="4">SCREEN</td></tr>
<tr><td>DEVICE</td><td colspan="4">device</td></tr>
<tr><td>WORKSTATION-WINDOW</td><td>left</td><td>right</td><td>bottom</td><td>top</td></tr>
</table>

PURPOSE:

Set display characteristics.

PARAMETERS:

COLOUR	Sets the output to the display device to be in colours (ON) or monochrome (OFF).
DESTINATION	Set the destination of the graphics produced in the DISPLAY command to the current plot file (FILE) or to the screen (SCREEN).
DEVICE	Set the current screen display device type. The available device types depend on the computer on which the program runs. Here is a selection of the some device types that may be available: VGA PC with VGA resolution, X-WINDOW for X windows, VT340 (Digital VT 340 screen), DUMMY used to dump display output to nowhere.
WORKSTATION-WINDOW	Set the size and position of the display window when using a workstation device. This command will only be taken into account if issued prior to any DISPLAY command. Otherwise, the settings will not be valid until the user has exited from Proban and entered again. Please note that the window can be re-sized using the mouse under X Windows.
left	Position of left display window border.
right	Position of right display window border.
bottom	Position of bottom display window border.
top	Position of top display window border.

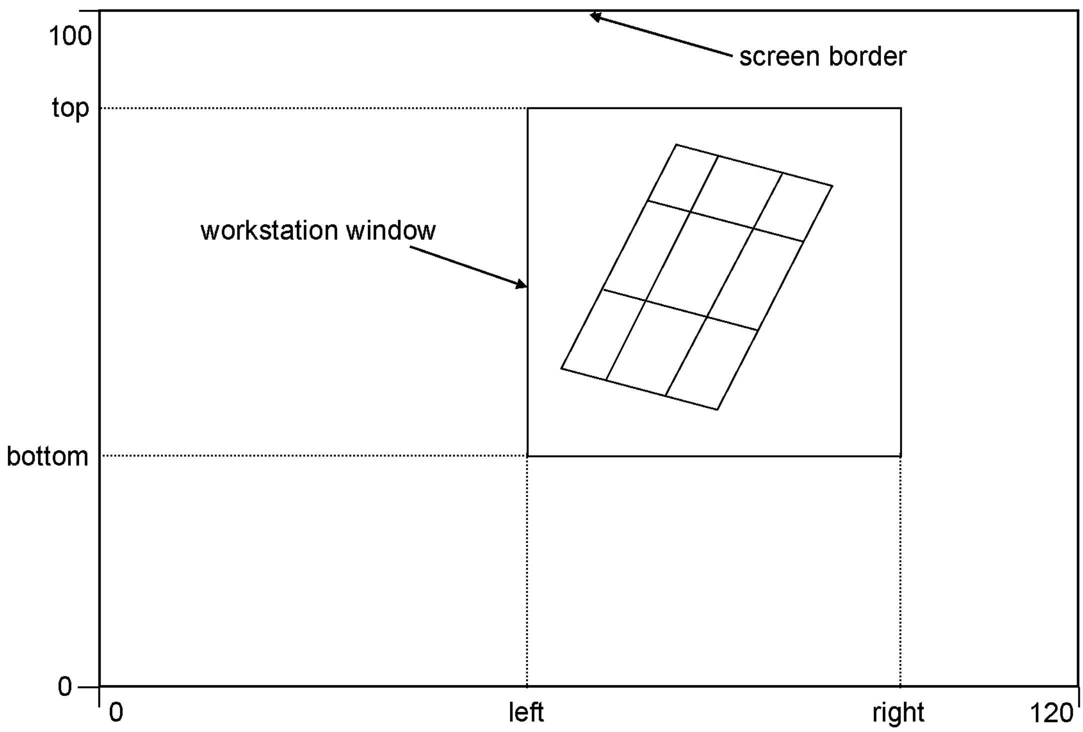


Figure 5.1 Setting the initial size of a workstation window

NOTES:

1 The destination is always set to SCREEN when the program starts up, also with an existing database.

2 The DUMMY device is useful for effectively disabling all DISPLAY commands in a command input file, when the displays themselves are not needed.

See also:

- DISPLAY
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET DISPLAY COLOUR ON
```

```
SET DISPLAY DESTINATION SCREEN
SET DISPLAY WORKSTATION-WINDOW <To be completed>
```

The default DEVICE depends on the computer system.

SET DRAWING

<table>
<tr><td rowspan="13">...</td><td rowspan="13">DRAWING</td><td rowspan="2">CHARACTER-TYPE</td><td colspan="2">HARDWARE</td></tr>
<tr><td colspan="2">SOFTWAREE</td></tr>
<tr><td rowspan="2">FONT-SIZE</td><td>ABSOLUTE</td><td>size</td></tr>
<tr><td>RELATIVE</td><td>factor</td></tr>
<tr><td rowspan="5">FONT-TYPE</td><td colspan="2">SIMPLE</td></tr>
<tr><td colspan="2">GROTESQUE</td></tr>
<tr><td colspan="2">ROMAN-NORMAL</td></tr>
<tr><td colspan="2">ROMAN-ITALIC</td></tr>
<tr><td colspan="2">ROMAN-BOLD</td></tr>
<tr><td rowspan="2">FRAME</td><td colspan="2">ON</td></tr>
<tr><td colspan="2">OFF</td></tr>
<tr><td rowspan="2">GRID</td><td colspan="2">ON</td></tr>
<tr><td colspan="2">OFF</td></tr>
</table>

PURPOSE:

To set drawing characteristics.

PARAMETERS:

CHARACTER-TYPE	Set the character type to SOFTWARE (i.e. scaleable) or HARDWARE (i.e. fixed).
FONT-SIZE	Set the font size. This affects all text.
ABSOLUTE size	Set the font size to an ABSOLUTE size in mm.
RELATIVE factor	Set the font size to a RELATIVE value scaleable by a factor, where 40*80 characters are fitted into the window when the factor is 1.
FONT-TYPE	Select the font to be used. The list of fonts may be machine dependent.
FRAME	Set frame on drawing ON or OFF. This command has currently no effect.
GRID	Set grid on a graph drawing ON or OFF.

NOTES:

See also:

- DISPLAY
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET DRAWING CHARACTER-TYPE SOFTWARE
SET DRAWING FONT-SIZE RELATIVE 1.0
SET DRAWING FONT-TYPE SIMPLE
SET DRAWING GRID ON
```

SET GRAPH

...	GRAPH	HISTOGRAM	...
		LINE-OPTIONS	...
		PIE-CHART	...
		XAXIS-ATTRIBUTES	...
		YAXIS-ATTRIBUTES	...
		ZAXIS-ATTRIBUTES	...

PURPOSE:

To set plot file characteristics.

PARAMETERS:

HISTOGRAM	Set options for display of a histogram.
LINE-OPTIONS	Set the options controlling how lines are drawn and marked.
PIE-CHART	Set options for display of a pie chart.
XAXIS-ATTRIBUTES	Set the options controlling the drawing and scale of the x-axis.
YAXIS-ATTRIBUTES	Set the options controlling the drawing and scale of the y-axis.
ZAXIS-ATTRIBUTES	Set the options controlling the drawing and scale of the z-axis.

NOTES:

All sub-commands and data are fully explained subsequently as each command is described in detail.

SET GRAPH HISTOGRAM

...	HISTOGRAM	COLUMNS	ncol
		FILLING	HATCHED
			HOLLOW
			SOLID

PURPOSE:

Set options controlling display of a histogram.

PARAMETERS:

COLUMNS ncol	Set the number ncol of columns in the histogram.
FILLING	The columns in the histogram can be filled with a HATCHED pattern, or not filled at all (HOLLOW), or be filled with a SOLID pattern.

NOTES:

1 To present a smooth histogram, the number of columns should be about 1/10 of the sample size or smaller.

2 When running the program on a black and white screen, it the it usually a good idea to change the default SOLID filling to a HOLLOW or HATCHED.

See also:

- DISPLAY RESULT DISTRIBUTION
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET GRAPH HISTOGRAM COLUMNS 20
SET GRAPH HISTOGRAM FILLING SOLID
```

SET GRAPH LINE-OPTIONS

<table>
<tr><td rowspan="5">...</td><td rowspan="5">LINE-OPTIONS</td><td>LINE-TYPE</td><td>line</td><td>linetype</td></tr>
<tr><td rowspan="2">MARKER</td><td colspan="2">ON</td></tr>
<tr><td colspan="2">OFF</td></tr>
<tr><td>MARKER-TYPE</td><td>line</td><td>marker type</td></tr>
<tr><td>MARKER-SIZE</td><td colspan="2">size</td></tr>
</table>

PURPOSE:

To set options controlling how lines are drawn and marked.

PARAMETERS:

LINE-TYPE	Controls how lines are drawn. Only six lines can be controlled.
line	A line number, from 1 to 6.
linetype	The line type to use. Legal values: BLANK, END-POINT, DASHED, DASH-DOT, DEFAULT, DOTTED, SOLID.
MARKER	Turn usage of markers ON or OFF.
MARKER-TYPE	Control the marker type.
marker type	The type of marker to use. Legal values: CROSS, DEFAULT, DELTA, DIAMOND, NABLA, PLUS, SQUARE, STAR
MARKER-SIZE size	Set the size of the markers.

NOTES:

Even when the MARKER option is ON, not all points on the curve need be marked. If more than 20 points are drawn and the line type is not BLANK, only a few points are marked in order to not clutter the curve with markers.

See also:

- DISPLAY
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET GRAPH LINE-OPTIONS LINE-TYPE DEFAULT % for all lines
SET GRAPH LINE-OPTIONS MARKER ON
```

```
SET GRAPH LINE-OPTIONS MARKER-TYPE DEFAULT % for all lines
SET GRAPH LINE-OPTIONS MARKER-SIZE 2.0
```

SET GRAPH PIE-CHART

<table>
<tr><td rowspan="14">...</td><td rowspan="14">PIE-CHART</td><td rowspan="2">EXPLODED-SEGMENT</td><td colspan="2">OFF</td></tr>
<tr><td>SEGMENT-NAME</td><td>name</td></tr>
<tr><td rowspan="3">FILLING</td><td colspan="2">HATCHED</td></tr>
<tr><td colspan="2">HOLLOW</td></tr>
<tr><td colspan="2">SOLID</td></tr>
<tr><td rowspan="9">LABEL</td><td rowspan="2">VISIBILITY</td><td>HIDE</td></tr>
<tr><td>SHOW</td></tr>
<tr><td rowspan="2">ORIENTATION</td><td>HORIZONTAL</td></tr>
<tr><td>ROTATED</td></tr>
<tr><td rowspan="3">POSITION</td><td>OUTSIDE</td></tr>
<tr><td>AUROMATIC</td></tr>
<tr><td>INSIDE</td></tr>
<tr><td rowspan="2">VALUE</td><td>ON</td></tr>
<tr><td>OFF</td></tr>
</table>

PURPOSE:

Set options controlling display of a pie chart.

PARAMETERS:

EXPLODED-SEGMENT	Controls if a segment of the pie is to be shown exploded (i.e. detached from the rest).
OFF	No segment is to be exploded.
SEGMENT-NAME name	Explode the segment with the given name. No segment will be exploded if the name does not match nay of the segment names in the pie to be displayed. The name can be abbreviated and the matching of names disregards the text case.
FILLING	The columns in the histogram can be filled with a HATCHED pattern, or not filled at all (HOLLOW), or be filled with a SOLID pattern.
LABEL	Define the drawing of the pie segment labels.
VISIBILITY	HIDE or SHOW the pie segment labels.

ORIENTATION	Draw the pie segment labels HORIZONTAL or ROTATED to follow the segment angle.
POSITION	Draw the pie segment labels OUTSIDE the pie, INSIDE the pie or use an AUTOMATIC placement, where they are drawn inside if possible.
VALUE	Show the value (size) of the pie segment (ON) or hide it (OFF).

NOTES:

When running the program on a black and white screen, it the it usually a good idea to change the default SOLID filling to a HOLLOW or HATCHED.

See also:

- DISPLAY RESULT IMPORTANCE-FACTORS
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET GRAPH PIE-CHART EXPLODED-SEGMENT OFF
SET GRAPH PIE-CHART FILLING SOLID
SET GRAPH PIE-CHART LABEL VISIBILITY SHOW
SET GRAPH PIE-CHART LABEL ORIENTATION HORIZONTAL
SET GRAPH PIE-CHART LABEL POSITION AUTOMATIC
SET GRAPH PIE-CHART LABEL VALUE ON
```

SET GRAPH XAXIS-ATTRIBUTES

<table>
<tr><td rowspan="10">...</td><td rowspan="10">XAXIS-ATTRIBUTES</td><td rowspan="4">DECIMAL-FORMAT</td><td colspan="3">EXPONENTIAL</td></tr>
<tr><td colspan="3">FIXED</td></tr>
<tr><td colspan="3">GENERAL</td></tr>
<tr><td colspan="3">INTEGER</td></tr>
<tr><td rowspan="2">LIMITS</td><td>FIXED</td><td>xmin</td><td>xmax</td></tr>
<tr><td colspan="3">FREE</td></tr>
<tr><td rowspan="2">SPACING</td><td colspan="3">LINEAR</td></tr>
<tr><td colspan="3">LOGARITHMIC</td></tr>
<tr><td rowspan="2">TITLE</td><td colspan="3">DEFAULT</td></tr>
<tr><td>SPECIFIED</td><td colspan="2">xtitle</td></tr>
</table>

PURPOSE:

Control the drawing of the X axis in a graph display.

PARAMETERS:

DECIMAL-FORMAT	Controls the presentation of numbers labelling the x axis.
EXPONENTIAL	The numbers are presented in exponential format (e.g. 1.233E+01).
FIXED	The numbers are presented in fixed format (e.g. 12.33).
GENERAL	The numbers are presented in general (free) format.
INTEGER	The numbers are presented as integers.
LIMITS	Controls the limits of the x axis.
FREE xmin xmax	The limits are determined by the data that are being presented.
FIXED	The limits are fixed to the minimum value xmin and the maximum value xmax.
SPACING	Controls the spacing of numbers along the axis.
LINEAR	The axis has a LINEAR spacing.
LOGARITHMIC	The axis has a logarithmic spacing with base 10.
TITLE	Set the title at the x axis.

DEFAULT	The title is specified by Proban according to the current graphs being drawn.
SPECIFIED xtitle	The specified xtitle text is used.

NOTES:

See also:

- DISPLAY
- PLOT
- SET GRAPH YAXIS-ATTRIBUTTES
- SET GRAPH ZAXIS-ATTRIBUTTES

EXAMPLES:

The following is default when the program starts with a new database:

```
SET GRAPH XAXIS-ATTRIBUTES DECIMAL-FORMAT GENERAL
SET GRAPH XAXIS-ATTRIBUTES LIMITS FREE
SET GRAPH XAXIS-ATTRIBUTES SPACING LINEAR
SET GRAPH XAXIS-ATTRIBUTES TITLE DEFAULT
```

SET GRAPH YAXIS-ATTRIBUTES

<table>
<tr><td rowspan="10">...</td><td rowspan="10">YAXIS-ATTRIBUTES</td><td rowspan="4">DECIMAL-FORMAT</td><td colspan="3">EXPONENTIAL</td></tr>
<tr><td colspan="3">FIXED</td></tr>
<tr><td colspan="3">GENERAL</td></tr>
<tr><td colspan="3">INTEGER</td></tr>
<tr><td rowspan="2">LIMITS</td><td>FIXED</td><td>ymin</td><td>ymax</td></tr>
<tr><td colspan="3">FREE</td></tr>
<tr><td rowspan="2">SPACING</td><td colspan="3">LINEAR</td></tr>
<tr><td colspan="3">LOGARITHMIC</td></tr>
<tr><td rowspan="2">TITLE</td><td colspan="3">DEFAULT</td></tr>
<tr><td>SPECIFIED</td><td colspan="2">ytitle</td></tr>
</table>

PURPOSE:

Control the drawing of the Y axis in a graph display.

PARAMETERS:

DECIMAL-FORMAT	Controls the presentation of numbers labelling the y axis.
EXPONENTIAL	The numbers are presented in exponential format (e.g. 1.233E+01).
FIXED	The numbers are presented in fixed format (e.g. 12.33).
GENERAL	The numbers are presented in general (free) format.
INTEGER	The numbers are presented as integers.
LIMITS	Controls the limits of the y axis.
FREE	The limits are determined by the data that are being presented.
FIXED ymin ymax	The limits are fixed to the min value ymin and the max value ymax.
SPACING	Controls the spacing of numbers along the axis.
LINEAR	The axis has a LINEAR spacing.
LOGARITHMIC	The axis has a logarithmic spacing with base 10.
TITLE	Set the title at the y axis.

DEFAULT	The title is specified by Proban according to the current graphs being drawn.
SPECIFIED ytitle	The specified ytitle text is used.

NOTES:

See also:

- DISPLAY
- PLOT
- SET GRAPH XAXIS-ATTRIBUTTES
- SET GRAPH ZAXIS-ATTRIBUTTES

EXAMPLES:

The following is default when the program starts with a new database:

```
SET GRAPH YAXIS-ATTRIBUTES DECIMAL-FORMAT GENERAL
SET GRAPH YAXIS-ATTRIBUTES LIMITS FREE
SET GRAPH YAXIS-ATTRIBUTES SPACING LINEAR
SET GRAPH YAXIS-ATTRIBUTES TITLE DEFAULT
```

SET GRAPH ZAXIS-ATTRIBUTES

<table>
<tr><td rowspan="10">...</td><td rowspan="10">ZAXIS-ATTRIBUTES</td><td rowspan="4">DECIMAL-FORMAT</td><td colspan="3">EXPONENTIAL</td></tr>
<tr><td colspan="3">FIXED</td></tr>
<tr><td colspan="3">GENERAL</td></tr>
<tr><td colspan="3">INTEGER</td></tr>
<tr><td rowspan="2">LIMITS</td><td>FIXED</td><td>zmin</td><td>zmax</td></tr>
<tr><td colspan="3">FREE</td></tr>
<tr><td rowspan="2">SPACING</td><td colspan="3">LINEAR</td></tr>
<tr><td colspan="3">LOGARITHMIC</td></tr>
<tr><td rowspan="2">TITLE</td><td colspan="3">DEFAULT</td></tr>
<tr><td>SPECIFIED</td><td colspan="2">ztitle</td></tr>
</table>

PURPOSE:

Control the drawing of the Z axis in a graph display.

PARAMETERS:

DECIMAL-FORMAT	Controls the presentation of numbers labelling the z axis.
EXPONENTIAL	The numbers are presented in exponential format (e.g. 1.233E+01).
FIXED	The numbers are presented in fixed format (e.g. 12.33).
GENERAL	The numbers are presented in general (free) format.
INTEGER	The numbers are presented as integers.
LIMITS	Controls the limits of the z axis.
FREE	The limits are determined by the data that are being presented.
FIXED zmin zmax	The limits are fixed to the min value zmin and the max value zmax.
SPACING	Controls the spacing of numbers along the axis.
LINEAR	The axis has a LINEAR spacing.
LOGARITHMIC	The axis has a logarithmic spacing with base 10.
TITLE	Set the title at the z axis.

DEFAULT	The title is specified by Proban according to the current graphs being drawn.
SPECIFIED ztitle	The specified ztitle text is used.

NOTES:

See also:

- DISPLAY
- PLOT
- SET GRAPH XAXIS-ATTRIBUTTES
- SET GRAPH YAXIS-ATTRIBUTTES

EXAMPLES:

The following is default when the program starts with a new database:

```
SET GRAPH ZAXIS-ATTRIBUTES DECIMAL-FORMAT GENERAL
SET GRAPH ZAXIS-ATTRIBUTES LIMITS FREE
SET GRAPH ZAXIS-ATTRIBUTES SPACING LINEAR
SET GRAPH ZAXIS-ATTRIBUTES TITLE DEFAULT
```

SET PLOT

...	PLOT	COLOUR	ON	
			OFF	
		FORMAT	SESAM-NEUTRAL	
			POSTSCRIPT	
			HPGL-7550	
			HPGL-2	
			CGM-BINARY	
		FILE	prefix	name
		PAGE-SIZE	A1	
			A2	
			A3	
			A4	
			A5	

PURPOSE:

To set plot file characteristics.

PARAMETERS:

COLOUR — Sets the output to the plot file to be in colours (ON) or monochrome (OFF).

FORMAT — Set the type of plot file to be used. Please note that the actual range of devices is machine dependent.

SESAM-NEUTRAL — SESAM Neutral format. This is the default format. It can be converted to other formats and/or manipulated by use if the utility program PLTCNV.

POSTSCRIPT — PostScript format (PostScript is a trademark of Adobe Systems Incorporated). Note that this requires access to a printer that accepts PostScript files.

HPGL-7550 — HP 7550 plotter.

HPGL-2 — HP Laserjet printer.

CGM-BINARY — ISO 8632-3 Computer Graphics Metafile (CGM) plot format.

FILE prefix name	Set the prefix and name of the plot file. The prefix and name are concatenated. The suffix of the file will depend on the format of the file.
PAGE-SIZE	Sets the size of the plot to one of A1, A2, A3, A4 or A5.

NOTES:

1 When one of these settings is changed, a new plot file will be opened the next time a plot is written.

2 One plot file may contain more than one plot.

3 There is two ways of generating a plot:

By use of the PLOT command

By use of SET DISPLAY DESTINATION FILE followed by a DISPLAY command

1 The CGM plot format is well suited for export of Proban plots to word processors such as Word, FrameMaker and DecWrite. You may transfer CGM files from one Operating System to another, just make sure to use the “binary” option when transferring the file with FTP (or another protocol).

2 Proban creates a new file each time you plot with the CGM format. Therefore you must specify a new name with each plot command. Otherwise you will overwrite the previous one. In Proban you may give a new plot file name with the command: SET PLOT FILE <prefix> <name>.

See also:

- SET DISPLAY DESTINATION
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET PLOT COLOUR ON
SET PLOT FILE % same prefix and name as the database and journal file
SET PLOT FORMAT SESAM-NEUTRAL
SET PAGE-SIZE A4
```

SET PRINT

<table>
<tr><td rowspan="6">...</td><td rowspan="2">DESTINATION</td><td colspan="2">FILE</td></tr>
<tr><td colspan="2">SCREEN</td></tr>
<tr><td>FILE</td><td>prefix</td><td>name</td></tr>
<tr><td rowspan="2">PAGE-ORIENTATION</td><td colspan="2">LANDSCAPE</td></tr>
<tr><td colspan="2">PORTRAIT</td></tr>
<tr><td>SCREEN-HEIGHT</td><td colspan="2">nlines</td></tr>
</table>

PURPOSE:

To set print characteristics.

PARAMETERS:

DESTINATION	Set the destination of the printed output to the SCREEN or to a FILE.
FILE prefix name	Set the prefix and name of the print file. The prefix and name are concatenated. The suffix of the file will be .LIS.
PAGE-ORIENTATION	Set the page orientation for the print file. See note 2 below:
LANDSCAPE	The print page is 132 characters wide.
PORTRAIT	The print page is 80 characters wide.
SCREEN-HEIGHT nlines	Set number of lines in one screen page to nlines. The purpose of this is to be able to pause the printout at the correct time when printing to SCREEN in a line mode run.

NOTES:

1 The print DESTINATION is reset to SCREEN each time Proban starts up, even if it is on an existing database.

2 The following figure illustrates the print layout:

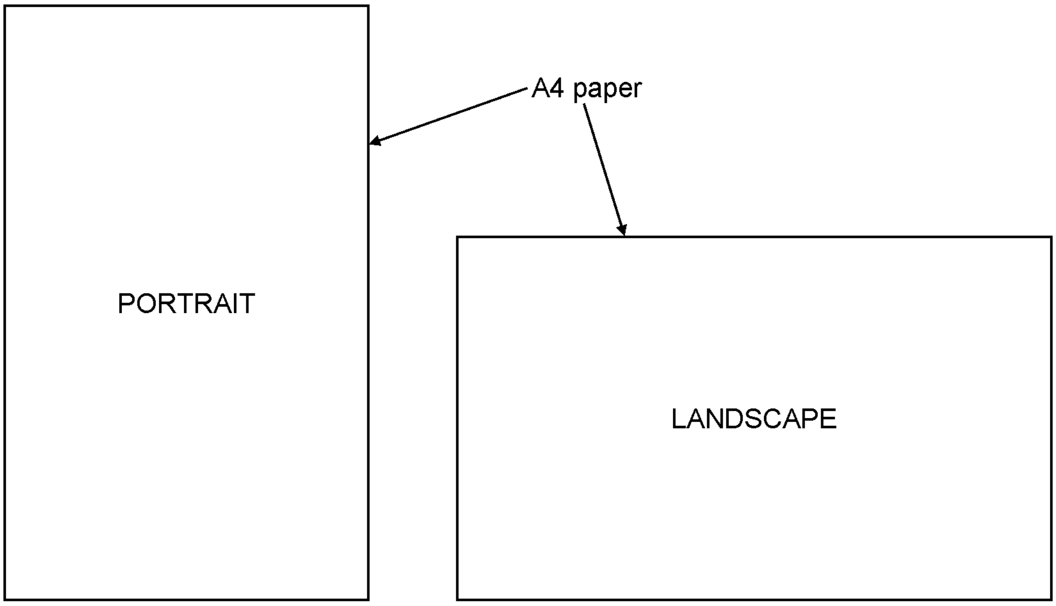


Figure 5.2 Setting PORTRAIT and LANDSCAPE print page orientation

See also:

- SET DISPLAY DESTINATION
- PLOT

EXAMPLES:

The following is default when the program starts with a new database:

```
SET PRINT DESTINATION SCREEN
SET PLOT FILE % same prefix and name as the database and journal file
SET PLOT PAGE-ORIENTATION LANDSCAPE
SET SCREEN-HEIGHT 24 % On VMS, Proban sets the correct height.
```

APPENDIX A PROBAN — LINK IN FUNCTIONS AND DISTRIBUTION

A 1 Implementing New Model Functions into Proban

How to program new model functions is described in Section 3.10.1 and Section 3.10.3.

A 1.1 Unix

Proban comes with a Makefile which can be copied and used to maintain a function library.

a In order to use this Makefile you should keep all source files of your function in one directory.

b First copy the Makefile from $SESAM_HOME/proban/funclb/Makefile to the directory where your model function routines are placed.

c Modify the Makefile. Follow the description in the Makefile itself.

d Usage of the Makefile to link Proban is described at the top of the Makefile itself. The commands must be typed from the directory where the Makefile is placed.

e To add a new model function, insert the file names in the definition of SOURCE in the Makefile, and then execute the make command.

A 1.2 NT

Proban comes with a set of batch files (.bat) and option files (.opt) which can be copied and used to maintain a function library.

a In order to use these files you should keep all source files of your function in one directory.

b First copy the .bat and .opt files from $SESAM_HOME/proban/funclb to the directory where your model function routines are placed.

c Modify the files. Follow the description in the files themselves.

d To add a new model function, insert the file names in the .opt files, and then execute the .bat files.

A 1.3 Implementing New Distributions into Proban

Each user may make his own library of distributions, extending the distributions library which is already available in Proban. How to do this is described in Section 3.9.3.

After the routines have been written, compiled and placed in the object library user.a (UNIX), the user should take a private copy of:

UNIX	Makefile, and modify it to contain the name and address of the new user.a.
NT	This facility is not available on NT.

Proban can the be linked with the new distribution.

It is not necessarily a trivial matter to include a new distribution into Proban, because it requires programming skills, and because Proban requires a very high accuracy of the inverse distribution function in the tails of the distribution. Please contact DNV Sesam AS if you need help to do this.